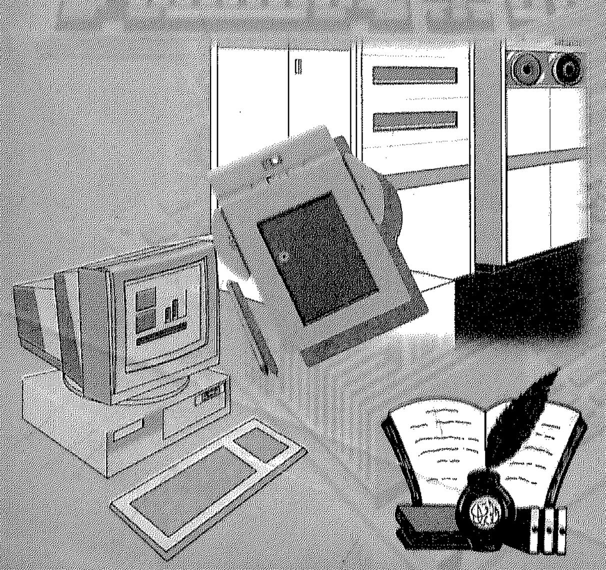
تحليل وتصميم أنظمة المعلومات الحاسوبية

Systems Analysis and Design

تأليف

د. محمد نور عبدالله برهان





بسم الله الرحمن الرحيم

تحليل وتصميم

أنظمة المعلومات الحاسوبية



تحليل وتصميم

أنظمة المعلومات الحاسوبية

System Analysis and Design

الدكتور

محمد نور عبد الله برهان

مؤسسة الوراق

للنشر والتوزيع

1998

رقم الإيداع لدى دائرة المكتبة الوطنية
(١٩٩٨/٦/٩٤٤)

رقم التصنيف : ٥٠٠ر٥

المؤلف ومن هو في حكمه : محمد نور عبدالله برهان

عنوان الكتاب : تحليل وتصميم انظمة المعلومات الحاسوبية

الموضوع الرئيسي : ١- المعارف العامة
٢- نظم البرمجة

بيانات النشر :

* تم إعداد بيانات الفهرسة والتصنيف الأولية من قبل دائرة الوطنية

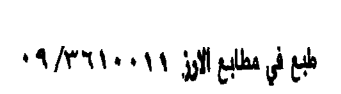
مؤسسة الوراق للنشر والتوزيع

عمان – شارع الجامعة – مقابل كلية الزراعة

تلفاكس 5337798

ص.ب 1527 عمان 11953





Converted by Tiff Combine - (no stamps are applied by registered version)

الفهرس المختصر

الصفحة

الجزء الرابع: الاتجاهات الحديثة في تحليل وتصميم الأنظمة

بسم الله الرحمن الرحيم

المقدمة

يشهد العالم اليوم، مع اقتراب دخوله إلى القرن الحادي والعشرين، نمواً كبيراً وانتشاراً واسعاً لأنظمة المعلومات الحاسوبية في مختلف مجالات الحياة. حيث تتزايد أعداد الحواسيب بمعدلات هائلة، وأضحت عنصراً أساسيا من عناصر العمل. ويرافق ذلك هذا السيل الجارف من تطبيقات تقنية الحاسوب والاتصالات في جميع مجالات الأعمال من إدارة وتعليم وصناعة وزراعة وصحة وبحث علمي وشؤون الأمن والدفاع والنقل وغيرها الكثير بما في ذلك الإعلام والثقافة والفنون والحياة المترلية.

لقد أدى انتشار تقنية المعلومات وتطبيقاتها على هذا النحو، إلى تزايد الاهتمام ببناء أنظمة معلومات حاسوبية ذات جودة عالية، تلبي متطلبات المستخدمين بكفاءة وفاعلية وبتكلفة مناسبة. ولقد ساعد ذلك في تطور علم تحليل وتصميم هذه الأنظمة بشكل سريع جداً خلال فترة زمنية قصيرة. فبالرغم من أنه لم يمضي على ظهور هذا الفرع العلمي الهام سوى أربعة عقود فقط، إلا أنه شهد وما يزال ظهور العديد من المنهجيات والأساليب والأدوات المختلفة. ويعود ذلك إلى التنوع الكبير في طبيعة الأنظمة المعلوماتية، وكذلك إلى تطور أدوات ووسائل وتقنيات بناء هذه الأنظمة .

وبالرغم من هذا التنوع الواسع والكبير في المنهجيات والأساليب والأدوات التي يمكن أن يستخدمها محلل ومصمم الأنظمة، إلا أن جوهر هذه العملية

يبقى في جميع الأحوال ، قائماً على فلسفة دراسة وتحليل المشكلات وإيجاد الحلول المناسبة لها ضمن القيود المتعلقة بالظروف المحيطة والإمكانات والموارد المتاحة. ولذلك فان جوهر عملية التحليل والتصميم هو فهم احتياجات المستخدمين وتحديدها بشكل واضح، ثم تحويلها إلى أهداف للنظام الجديد، وصياغتها بشكل يمكن قياسه. يلي ذلك البحث عن البدائل المختلفة التي يمكن من خلالها تلبية هذه الاحتياجات. ولذلك كان تركيزنا في هذا الكتاب على تعريف الطالب وتمكينه من فهم جوهر عملية التحليل والتصميم ليكون قادراً على الاستفادة من المنهجيات والأساليب المتنوعة، واختيار ما يناسب كل نظام من الأنظمة التي يقوم بتحليلها وتصميمها. ويعتمد هذا الكتاب بشكل عام على استخدام المدخل النظمي أو الشمولي System Approach لتحليل وتصميم أنظمة المعلومات بشكل مهيكل Structured Analysis and Design.

يتضمن هذا الكتاب ستة عشر فصلاً موزعة إلى أربعة أجزاء رئيسية. يدرس الجزء الأول وهو بعنوان "مقدمة إلى تحليل وتصميم أنظمة المعلومات الحاسوبية" المفاهيم الأساسية لهذه الأنظمة وتطبيقاتها المختلفة في المنظمة، والأساليب أو المداخل المختلفة لتطويرها. أما الجزء الثاني وعنوانه " تحليل أنظمة المعلومات الحاسوبية" فيدرس المراحل الثلاث الأولى في دورة حياة تطوير هذه الأنظمة، بدءاً من تجميع المعلومات والحقائق وانتهاء ببناء النموذج المنطقي للنظام الحالي، وتحديد متطلبات تطويره. ويدرس الجزء الثالث وعنوانه " تصميم أنظمة المعلومات الحاسوبية" مرحلة التصميم، وهذا يشمل جميع الأنشطة والعمليات التي يتم من خلالها التوصل إلى النموذج المادي للنظام الجديد، وإعداد التصاميم المتعلقة بمكوناته: واجهات الاستخدام وقاعدة البيانات والبرامج اللازمة لتنفيذه وفقا للأهداف التي

تم تحديدها في مرحلة التحليل. وأخيرا يأتي الجزء الرابع وهو بعنوان " الاتجاهات الحديثة في تحليل وتصميم الأنظمة" ، ليدرس أهم المنهجيات الحديثة المستخدمة في التحليل والتصميم كمنهجية التطوير الموجه للكيانات Object-Oriented (OOD) Development، وهندسة البرمجيات بمساعدة الحاسوب Computer-Aided Software Engineering(CASE)، ومنهجية هندسة المعلومات Information Engineering وغيرها. وينتهي هذا الجزء الأخير بدارسة أساليب إدارة المشروعات ودورها في تخطيط ومتابعة تنفيذ أنظمة المعلومات، لتتم ضمن معايير الجودة المقررة وبتكلفة معقولة، وبأقصر زمن ممكن.

وفي الختام آمل أن يساعد هذا الكتاب طلابنا الأعزاء الذين يدرسون التخصصات الحاسوبية في الجامعات والمعاهد العليا، وبشكل خاص أولئك الذين يجدون صعوبة كبيرة في استخدام المراجع الأجنبية، كما يمكن أن يفيد هذا الكتاب طلبة إدارة الأعمال والمحاسبة، باعتباره مرجعا ضروريا لدراسة وفهم أنظمة المعلومات الإدارية والمحاسبية وتطبيقاتها المختلفة.

والله ولي التوفيق

الدكتور محمد نور عبد الله برهان



مقدّمة إلى تحليل وتصميم أنظمة المعلومات الحاسوبية



الفصل الأول

المفاهيم الأساسية في تحليل وتصميم أنظمة المعلومات

1- المقدمة

يزداد الاهتمام بالمعلومات وأنظمتها لكونها مورداً استراتيجياً لمعظم منظمات الأعمال في المجتمع المعلوماتي الحديث. فهي توفر الدعم اللازم لتحقيق مزايا تنافسية تساعد في تقوية موقع المنظمة وتضمن إستمراريتها ونجاحها. وهذا لا يتحقق بشكل آلي، بل يتطلب من المنظمات العمل على تطوير أنظمة المعلومات اللازمة لها، والتي تمكنها من رفع مستوى أداء عملياتها وتحسين إنتاجيتها، وزيادة رضى زبائنها وتقوية معنويات موظفيها. ويقصد بتطوير أنظمة المعلومات، تخطيط وتصميم وتنفيذ هذه الأنظمة، ويدرس هذا الكتاب بشكل مفصل أهم مرحلتين في عملية التطوير هذه وهما:

مرحلة التحليل والتي يتم خلالها تحديد احتياجات المنظمة ومتطلباتها المتعلقة بنظام المعلومات الذي يجري تطويره، ومرحلة التصميم التي يتم فيها إعداد التصاميم التفصيلية لتنفيذ هذه الأنظمة بشكل يلبي الاحتياجات التي تم تحديدها في المرحلة السابقة بكفاءة وفاعلية.

أما هذا الفصل فهو يشرح المفاهيم الأساسية في تحليل النظم وتصميمها، ويركز على التعريف بمفهوم النظام بشكل عام وتحديد مكوناته وخصائصه، ثم ينتقل إلى تعريف أنظمة المعلومات الحاسوبية وتحديد أهدافها ومكوناتها وهيكليتها ومتطلبات تطويرها، وفق معايير الجودة والفاعلية وسهولة الاستخدام والصيانة

وغيرها. ويقصد بالنظام عادة **مجموعة مكونات مرتبطة ببعضها البعض للقيام بوظيفة معينة** تكون عادة هدف النظام، فأي نظام يتكون من أفراد وتجهيزات وإجراءات ومواد تتفاعل معاً لتحقيق غرض معين. فالشركة الصناعية هي نظام تتفاعل فيه مكونات عديدة وهي: العمال والإدارة والآلات والمواد الأولية وغيرها، بهدف إنتاج المنتجات المطلوبة. والجامعة أيضاً نظام يتكون من الطلاب والمدرسين والمناهج الدراسية والوسائل المتنوعة التي تتفاعل معاً لإعداد الكوادر وتأهيلها في الاختصاصات المختلفة. وسنتعامل من خلال دراسة هذا الكتاب مع نوعين من الأنظمة هما:

- **أنظمة الأعمال Business Systems**: ويقصد بها أنظمة العمليات الجارية في مختلف منظمات الأعمال (مصانع-بنوك-جامعات-إدارات حكومية وغيرها) وتهتم هذه الأنظمة بتحقيق أهداف محددة في مجالات الأعمال التي تمارسها. ويمكن دراسة هذه الأنظمة في عدة مستويات: المستوى الكلي للنظام، ثم مستوى الأنظمة الفرعية المكونة له كأنظمة الإنتاج والتسويق والمبيعات والمشتريات، والتي تعتبر من الأنظمة الفرعية للنظام الكلي. كما يمكن الاستمرار في تقسيم هذه الأنظمة الفرعية إلى أنظمة فرعية تابعة لها وهكذا.
- **أنظمة المعلومات Information Systems** : وتهتم بإدارة البيانات وتوفير المعلومات اللازمة لتسيير أنظمة الأعمال المشار إليها أعلاه، وتتكون أنظمة المعلومات من الأفراد والإجراءات والتقنيات Technology والبيانات، وكل هذه العناصر تتفاعل معاً لتزويد الإدارة بالمعلومات اللازمة لتسيير أنظمة العمل المختلفة. وبذلك

فإن هيكلية أنظمة المعلومات ومستوياها تتوافق تماماً مع مستويات وهيكلية أنظمة الأعمال المرتبطة بها.

2- الخصائص العامة للأنظمة:

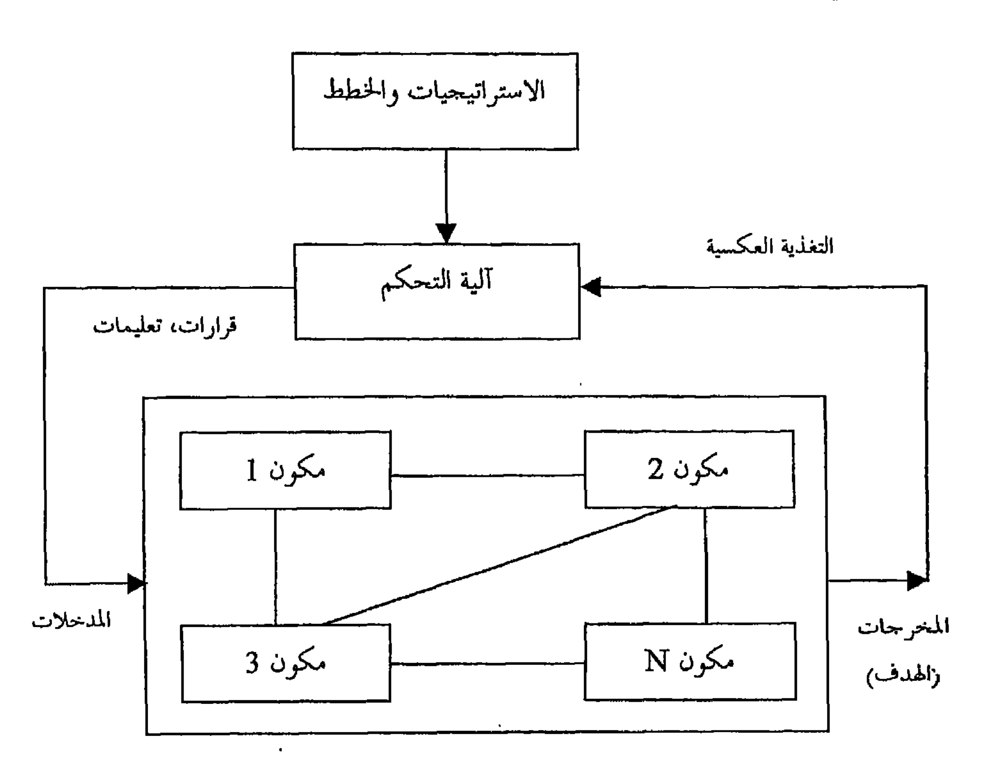
يعرف النظام System بأنه " **مجموعة من العناصر أو المكونات التي تتفاعل مع بعضها البعض لتحقيق هدف محدد**".

يتبين من هذا التعريف العام للنظام أن :

- للنظام هيكلية هرمية محددة، أي يتكون من مجموعة من المكونات أو العناصر والتي تمثل بدورها أنظمة فرعية**Subsystems**. وكذلك غالباً ما يكون النظام نفسه أحد مكونات نظام أكبر.فإذا اعتبرنا الكلية في الجامعة نظاماً يتكون من الأقسام العلمية، فإن الكلية نفسها هي جزء أو مكون من نظام أكبر وهو الجامعة التي تعتبر بدورها جزءاً من نظام أكبر وهو نظام التعليم العالي في الدولة.
- تتفاعل مكونات النظام مع بعضها البعض وهذا يعني أنها لا تعمل بشكل مستقل ومنفصل بل **تعمل معاً** لتحقيق هدف النظام، وبما أن أي عمل جماعي يتطلب تنسيقاً محكماً ودقيقاً فإن النظام يجب أن تتوفر فيه إمكانية الضبط والتحكم في مكوناته المختلفة.
- للنظام **هدف** محدد، أي لا يوجد نظام بدون هدف، ويتم تحقيق هذا الهدف من خلال قيام النظام بالوظيفة أو الوظائف المطلوبة منه. ويكون هذا الهدف هو تلبية احتياجات محددة في بيئة النظام، لذلك

فإن مبرر وجود النظام (الهدف) هو تلبية هذه الاحتياجات المتواجدة في البيئة، وهكذا فإن النظام يوجد دوما ويعمل في إطار بيئة محددة.

نستنتج مما سبق الخصائص أو المبادئ **System Characteristics** العامة للنظم والتي يمكن تلخيصها على النحو التالي: (انظر الشكل 1.1).



بيئة النظام

شكل (1.1) الخصائص الأساسية للنظام

1.2- الهدف Goal : لكل نظام هدف أو أهداف تحدد بشكل دقيق ما يجب أن يقوم به النظام، فمثلاً هدف الجامعة تعليم الطلاب وتأهيلهم ليصبحوا مهندسين أو أطباء أو مدراء أو غير ذلك.

2.2- بيئة النظام System Environment:

يهدف النظام عادة إلى تلبية حاجة موجودة في بيئة النظام. ويقصد بالبيئة مجموعة العوامل الموجودة خارج حدود النظام والتي تؤثر في سلوك النظام وظروف عمله، ويستمد منها مدخلاته ويقدم إليها مخرجاته.

3.2- حدود النظام System Boundary:

هي الإطار الذي يضم جميع مكونات النظام، فكل ما يقع خارج هذا الإطار لا ينتمي إلى النظام بل إلى البيئة التي يعمل فيها. وكل ما يوجد داخل إطار النظام يعتبر من مكوناته.

4.2- المكونات أو النظم الفرعية: Subsystems

يتكون النظام من عدد من المكونات أو النظم الفرعية، ويقوم كل نظام فرعي أو مكون بأداء وظيفة محددة تكون جزء من الوظيفة العامة للنظام. وترتبط الأنظمة الفرعية معاً من خلال التدفقات المختلفة **Flows** فيما بينها، والنظام الجيد هو ذلك النظام الذي تكون فيه هذه التدفقات أقل ما يمكن **Minimal**. أي تكون مكوناته ذات استقلالية كبيرة نسبياً، وبالمقابل عندما يكون عدد الروابط

(التدفقات) كبيراً يصبح النظام أكثر تعقيداً وتتزايد إعتمادية مكوناته على بعضها البعض.

وتحدد مكونات النظام ما يسمى هيكل النظام **System Structure**. وتتم دراسة هيكل النظام من خلال تقسيم مكوناته بحسب وظائفها إلى مكونات تتعامل مع المدخلات وأخرى تقوم بإنتاج المخرجات وثالثة تقوم بعمليات المعالجة والتخزين في النظام.

5.2- التغذية العكسية Feedback:

يجب أن يوفر النظام تدفق بيانات التغذية العكسية إلى آلية أو وظيفة التحكم لتمكينها من ضبط أداء النظام، وتعكس هذه البيانات الأداء الفعلي لعمليات النظام.

6.2- آلية التحكم Control Mechanism:

وتقوم بوظيفة ضبط مدخلات وعمليات النظام من خلال مقارنة بيانات التغذية العكسية (المخرجات الفعلية للنظام) مع الأهداف المنشودة وتحديد الانحرافات واتخاذ الإجراءات المناسبة بشأنها.

وأخيراً تجدر الإشارة إلى أنه يمكن تصنيف الأنظمة بحسب معايير عديدة أهمها:

أ- **درجة التعقيد**: ويقصد بها عدد العناصر التي يتكون منها النظام والعلاقات التي تربط فيما بينها. **فالنظم البسيطة** تتكون عادة من عدة عناصر تتمتع باستقلال نسبي كبير وترتبط معاً بأقل عدد ممكن من

العلاقات، **أما النظم المعقدة** فهي تلك التي تتكون من مجموعة كبـــيرة من العناصر التي ترتبط معاً بعلاقات عديدة ومتشابكة.

ب- **طبيعة النظام**: تصنف النظم إلى **أنظمة طبيعة (ماديـــة) Physical** كالسيارة والمصنع والطائرة وجسم الإنسان وغيرها، **ونظم مفاهيميـــة أو منطقية Conceptual** كالنظم الاجتماعية أو الثقافيـــة. وتنتمـــي أنظمة المعلومات إلى هذا النوع من النظم المفاهيمية.

جـ- **طريقة صنع النظام**: تصنف الأنظمة وفقـــاً لطريقـــة صنعـــها أو مصدرها إلى **أنظمة طبيعية Natural Systems** من صنــــع الله سبحانـــه وتعالى كالإنسان والكرة الأرضية والنظام الكوني والمجموعـــة الشمســـية وغيرها، وإلى **أنظمة من صنع الإنسان** كالآلات والحواسيب والبرمجيـــات وأنظمة المعلومات الحاسوبية وغيرها.

د- **علاقة النظام بالبيئة التي يوجد فيها**: وهو المعيار الأهم لتصنيف النظم، حيث يمكن أن يكون النظام.

- **مغلقاً Closed System** أي لا يتأثر ولا يؤثر في البيئة التي يوجد فيـــها، وفي هذه الحالة يكون التحكم بهذا النظام داخلياً وتلقائياً بحسب طبيعة البيانات الـــتي يولدها، وهذا النوع من الأنظمة يندر وجودها عملياً.
- **مفتوحاً Open System**: أي يتفاعل مع البيئة الموجود فيها مـــن خـــلال مدخلاته التي يحصل عليها من هذه البيئة ومخرجاته التي يقدمها إليها.

3. أنظمة المعلومات الحاسوبية :

1.3 تعريف : في ضوء التعريف السابق للنظام يمكن تعريف أنظمة المعلومات الحاسوبية بأنها **مجموعة الأفراد والعمليات والبيانات والتقنيات التي تتفاعل معاً لتزويد المدراء أو الموظفين بالمعلومات اللازمة لهم** . وهذا يعني :

- **الهدف Goal** : يهدف نظام المعلومات الحاسوبية إلى توفير المعلومات اللازمة لتسيير وإدارة العمل في المنظمات المختلفة .
- **الحدود Boundary** : تشمل أنظمة المعلومات الحاسوبية جميع الأنشطة والعمليات المتعلقة بالتقاط البيانات من مصادرها ونقلها وتخزينها ومعالجتها وتوفيرها للمستخدمين بالشكل والوقت والحجم المناسبين .
- **البيئة Environment** : بيئة أنظمة المعلومات الحاسوبية هي المنظمة التي يقوم النظام بتوفير المعلومات اللازمة لإدارتها .

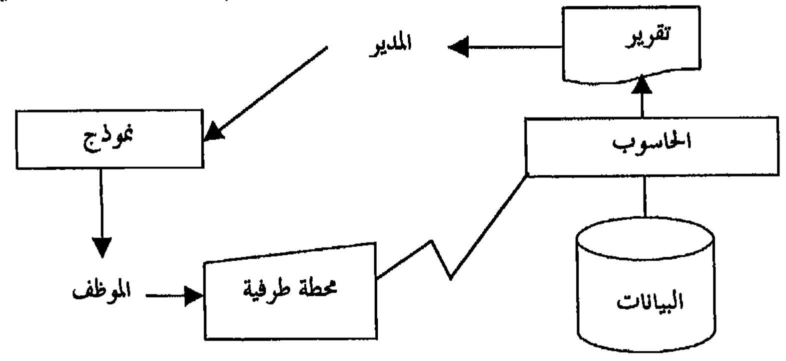
ويقصد بكلمة **منظمة Organization** ، التي سنستخدمها كثيراً في هذا الكتاب ، أية شركة أو مؤسسة أو هيئة سواء كانت رسمية (حكومية) أو خاصة ، تقوم بالنشاطات الاقتصادية أو الإجتماعية ، تهدف للربح أو غير هادفة له ، ومهما كان مستواها التنظيمي. وبذلك فإن مصطلح المنظمة يشير بشكل عام إلى أي من المسميات التالية :

مدرسة – جامعة – وزارة – هيئة حكومية – معمل – شركة صناعية – متجر – مستودع – شركة تجارية – مكتب هندسي – صيدلية – مستشفى أو غير ذلك .

2.3 المكونات : يتكون نظام المعلومات الإدارية من :

- **الأفراد People** : وهذا يشمل المستخدمين والمدراء وأخصائيي أنظمة المعلومات، أي كل الأفراد الذين يقومون باستثمار النظام أو تطويره .
- **العمليات Processes** : ويقصد بها مجموعة الخطوات والأنشطة التي يتم من خلالها التقاط وتجميع ونقل وتخزين ومعالجة المعلومات وتقديمها إلى المستفيدين منها .
- **التقنية Technology** : وهي التنقية والتسهيلات والمعدات المستخدمة لإنجاز العمليات المذكورة أعلاه ، وهذا يشمل الحواسيب وأجهزتها الطرفية وكذلك وسائل وأجهزة الاتصالات والبرمجيات وغيرها .
- **البيانات Data** : وهي المادة الخام التي تعتبر أساس عمل نظام المعلومات ومبرر وجوده. تنشأ البيانات من خلال الأحداث المختلفة التي تقع داخل وخارج المنظمة **Internal and external Transactions** ، ويعمل النظام على التقاط وتجميع هذه البيانات وتخزينها ومعالجتها لتحويلها إلى معلومات يتم توزيعها على المستفيدين منها .

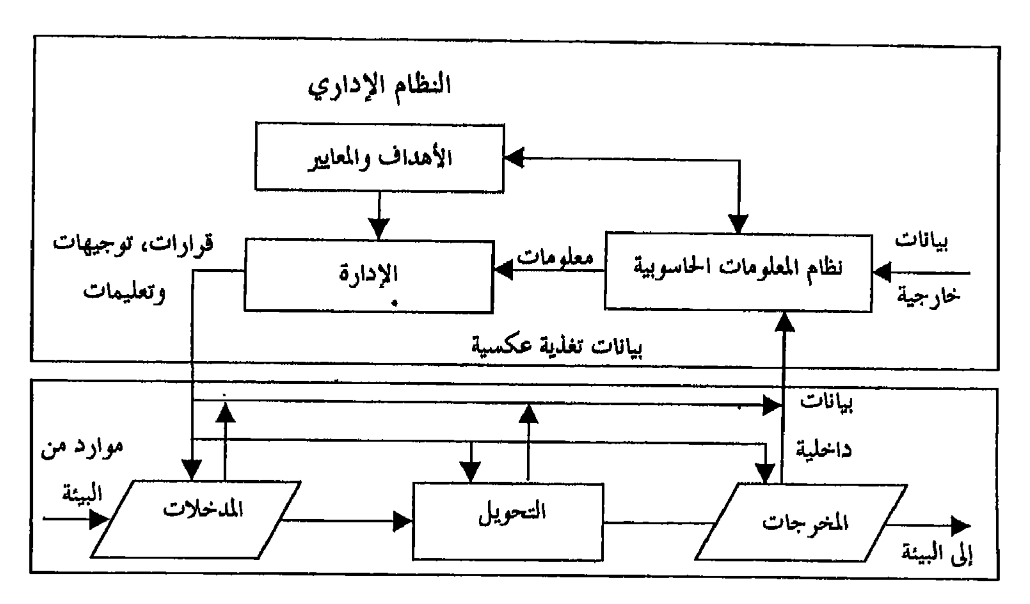
ويبين الشكل (2.1) التالي مخططاً رمزياً لمكونات نظام المعلومات الحاسوبي.



شكل (2.1): المكونات الأساسية لنظم المعلومات الحاسوبية

3.3 بيئة نظام المعلومات الحاسوبية :

يهدف نظام المعلومات إلى دعم عمل الأفراد العاملين في المنظمة (المستخدمين : مدراء وموظفين) من خلال توفير المعلومات اللازمة لهم بالحجم والشكل المناسبين وفي التوقيت المناسب . ومن خلال ذلك يتضح أن بيئة نظام المعلومات هي المنظمة التي يتواجد في إطارها . فنظام المعلومات يقوم بتجميع البيانات المتعلقة بسير العمل في المنظمة بهدف معالجتها وتخزينها واستخدامها لمقارنة مستويات الأداء الفعلية مع المستويات المخططة أو المستهدفة ، ووضعها أمام إدارة المنظمة بشكل تقارير بالإستثناءات **Exceptional reports** تساعدها في التحكم وضبط سير العمليات المختلفة باتجاه الأهداف المنشودة . ويوضح الشكل (3.1) علاقة نظام المعلومات ببيئته الخارجية أي بكل من النظام المادي للمنظمة و النظام الإداري فيها.



شكل (3.1) نظام المعلومات الحاسوبية وعلاقته بالنظام المادي والنظام الإداري في المنظمة

وهكذا نجد أن نظام المعلومات الحاسوبيه هو نظام مفاهيمي مفتوح كما هو مبين في الشكل (3.1) ويحصل على البيانات من البيئة : النظام المادي للمنظمة (بيانات داخلية) وبيئة عمل المنظمة ، ويقدم مخرجاته إلي إدارة المنظمة لمساعدتها في تسيير العمليات واتخاذ القرارات وبشكل عام تحقيق أهداف المنظمة بالكفاءة والفاعلية المطلوبة .

4. عملية تطوير أنظمة المعلومات الحاسوبية

تهدف عملية تطوير نظم المعلومات الحاسوبية إلى بناء أنظمة معلومات تقوم بدعم عمل الأفراد في المنظمات في جميع مستوياتها ، لتمكينهم من إنجاز الأعمال وحل المشكلات وإتخاذ القرارات .

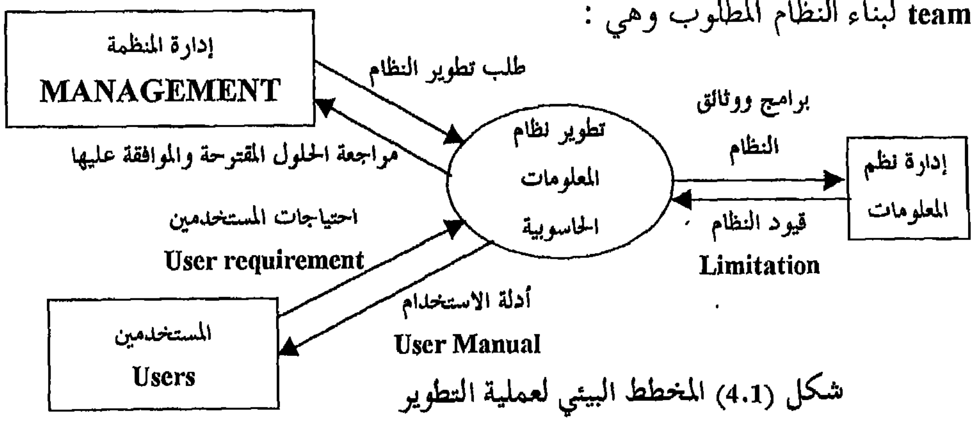
وتتضمن عملية تطوير أنظمة المعلومات الحاسوبية ثلاث مراحل أساسية هي :

- **مرحلة التحليل Analysis** : يتم خلالها دراسة النظام الحالي وفهم مكوناته وعملياته والمشاكل التي تواجهه ، و القصورات الموجودة فيه . وتنتهي عملية التحليل بتحديد إحتياجات المستخدم **User Requirements** التي تصف ما يرغبه المستخدم أو يتوقعه من النظام .
- **مرحلة التصميم Design** : ويتم خلالها إيجاد التصميم الأمثل لنظام المعلومات الحاسوبي الذي يلبي إحتياجات المستخدمين التي تم توصيفها في مرحلة التحليل .
- **مرحلة البناء Building** : ويتم خلالها تنفيذ التصميم الذي تم التوصل إليه في المرحلة السابقة ،وتحويله إلى نظام جاهز للعمل . وقد تتضمن هذه

المرحلة بناء نظام جديد تماما ،أو تعديل نظام المعلومات الحالي بإجراء بعض التغييرات فيه .

وتتألف المراحل الثلاث المذكورة أعلاه من عدد من الخطوات التي تقسم بدورها إلى أنشطة ومهام . ويتطلب القيام بهذه الأنشطة استخدام العديد من الأساليب وأدوات التحليل والتصميم . ولقد تطورت عملية التحليل والتصميم بشكل سريع خلال السنوات الماضية وظهرت أساليب وتقنيات جديدة تساعد في تسريع عملية التطوير وزيادة جودتها وإنتاجيتها . ونتيجة لذلك ظهرت العديد من الأدوات المستخدمة في نمذجة العمليات وقواعد البيانات كمخططات تدفق البيانات والنماذج التجريبية **Prototyping** وهندسة المعلومات وهندسة البرمجيات بمساعدة الحاسوب وغيرها. كما ظهرت منهجيات جديدة لعملية التطوير سنقوم بدراستها والتعرف عليها جميعا في الفصول القادمة من هذا الكتاب .

يتم تطوير نظم المعلومات الحاسوبية عادة من خلال جهد جماعي مشترك يقوم به فريق تطوير متكامل يضم بالإضافة إلي أخصائيي أنظمة المعلومات ممثلي الفئات المختلفة للمستفيدين من النظام من مدراء أو موظفين . ويبين الشكل (4.1) بيئة هذه العمليه والجهات التي يجب أن يتعاون معها فريق التطوير **Development team** لبناء النظام المطلوب وهي :



شكل (4.1) المخطط البيئي لعملية التطوير

أ) **إدارة المنظمة** : التي سيتم فيها تطوير النظام المطلوب ، والتي من المفترض أن يساهم النظام في تحسين أداء عملياتها، وتتحمل التكاليف الناجمة عن تطويره وإدامته .

ب) **المستخدمين** أو المستفيدين من النظام **Users** : ويقصد بهم جميع الأفراد الذين يحتمل أن يستخدموا النظام أو يستفيدوا من مخرجاته أو يتأثر به عملهم .

جـ) **إدارة نظم المعلومات** : التي ستوفر التسهيلات اللازمة لعمل النظام من تجهيزات حاسوبية وأجهزة طرفية وشبكات نقل بيانات وبرمجيات حاسوبية مختلفة ، بالإضافة إلى قيامها بالعمليات اللازمة لإدامة النظام وحمايته .

5. أهداف وأهمية تحليل وتصميم أنظمة المعلومات

تهدف عملية التحليل والتصميم إلى بناء أنظمة معلومات حاسوبية تلبي إحتياجات محددة للمنظمة . وتنشأ الحاجة إلى هذه العملية في الحالات التالية :

- وجود مشاكل في نظام المعلومات الحالي تجعله غير مناسباً (تكلفة عالية – إجراءات بطيئة – نقص في المعلومات وغيرها) . مما يتطلب إجراء تغييرات في النظام الحالي ، أو استبداله بشكل كامل .
- الاستفادة من الفرص التي يتيحها استخدام تقنية المعلومات في تحسين الأداء في المنظمة وتحقيق مزايا تنافسية .
- تحسين النظام الحالي بهدف تحسين الأداء أو زيادة الإنتاجية أو تحقيق مزايا تنافسية للمنظمة .

- ظهور احتياجات جديده في المنظمة مما يتطلب تعديل نظام المعلومات الحالي ، بالرغم من عدم وجود قصورا في عمله ، ليتمكن من تلبية هـذه الاحتياجات الجديدة .
- ظهور تقنيات جديده يمكن أن تساهم في تحديث أنظمة المعلومات الحالية وتساعد في تقليل التكلفة أو تحسين مخرجات هذه الأنظمة ، أو توفير خدمات معلوماتية أفضل .
- تطبيق تعليمات أو توجيهات إدارية عليا لتوفير معلومات معينة وتزويد الجهات ذات العلاقة بها (مؤسسات حكومية – مجلس الإدارة – هيئة المؤسسين – حملة الأسهم وغيرها) .

وبشكل عام فإن تحليل وتصميم أنظمة المعلومات يجب أن يسعى إلى تحقيق مجموعة من الأهداف التالية :

- **توفير إمكانات افضل Better Capabilities** لأداء العمليات المختلفة في المنظمة . وهذا يعني :
  - إنجاز العمليات بطريقة أسرع .
  - معالجة كميات كبيره من المعلومات بوقت قصير جداً .
  - الاسترجاع السريع للمعلومات .
- **توفير تحكم أفضل Better Control** :
  - إنجاز العمليات بدقه عاليه .
  - إنسجامية كبيره في العمليات .
  - تحسين إجراءات الأمن والحماية .

- **تحسين الاتصالات وتدفق المعلومـــات Improve Communication** بين مختلف مواقع العمل في المنظمة .
- **تخفيض تكاليف العمليات** والتحكم فيها لتكون ضمـــن المســتويات المقررة .
- **توفير مزايا تنافسية** تدعم الموقـــف التنافســي للمنظمـــة وتضمـــن إستمراريتها ونجاحها . وذلك من خلال التوجه نحو الزبائن وتحسين العلاقـــات مع الموردين وغير ذلك من الإمكانيات الــــتي يوفرهـــا اســتخدام التقنيـــات المعلومـــاتية الحديثة .

وتعتبر عملية التحليل والتصميم نشاط ضروري لا غنى عنه عند بناء أنظمـة معلومات جديده أو تحسين الأنظمه الموجوده ، فمن خلال هـــذه العمليـــة يمكـــن تحقيـــق ما يلي :

- فهم الأنظمة الحالية والتعرف على مشاكلها .
- تحديد الاحتياجات المعلوماتية .
- تصميم الإجراءات والعمليات اللازمة لتلبية هذه الاحتياجات .
- اختيار التقنية المناسبة من تجهيزات وبرمجيات لبناء هذه الأنظمة .
- مراجعة وتدقيق الحلول المقترحة للتأكد من جودتها أولاً، ثم للتأكد من تنفيذ النظام وفقاً للتصاميم المقترحة وبما يلبي معايير الأداء المقررة .

6. الأسلوب الهيكلي وتصميم أنظمة المعلومات

إن الغاية الأساسية من تحليل وتصميم الأنظمة هي تطوير أنظمة معلومـــات حاسوبية تتميز بما يلي :

- **الفعالية Effectiveness** : ويقصد بها تلبية إحتياجات المستخدمين والمنظمة والجهات التي تتعامل معها .
- **الكفاءة Efficiency** : وتعني الاستخدام الأمثل للموارد المتاحة .
- **سهولة الاستخدام Usability** : أي تتوفر فيها جميع الإمكانات التي تجعلها سهلة الاستخدام .
- **الوثوقية Reliability** : ويقصد بها القدرة على العمل في مختلف الظروف وبدون أعطال .
- **سهولة الصيانة Maintainability** : وتعني إمكانية إدخال التعديلات المختلفة فيها بأقل تكلفه ممكنة .

وللوصول إلى أنظمة معلومات حاسوبيه تحقق الأهداف المذكورة أعلاه يستخدم الأسلوب الهيكلي **Structured approach** لتحليل وتصميم الأنظمة والذي يوفر طريقة منظمة **Systematic way** لتنفيذ هذه المهام .

يقوم المدخل الهيكلي على المبادئ التالية :

1) **تقسيم المشاكل** **Decomposition** أو الأنظمة التي يجري تحليلها وتصميمها إلى عدد من الأجزاء المكونة لها ، مما يجعل دراسة هذه المشاكل أو الأنظمة أكثر سهوله . إن تقسيم النظام إلى مكوناته يجعل إدارة عملية التطوير ،بشكل عام ،تتم بصورة أفضل .

2) **استخدام مدخل من الأعلى نحو الأسفل Top–down approach** للتعامل مع المشكلة التي تجري دراستها . حيث يبدأ التحليل من النظرة الشاملة إلى النظام ككل : مكوناته (وظائفه الأساسيه) والعلاقات التي تربط فيما بينها، ثم دراسة وتحليل كل مكون (كل

وظيفة رئيسيه) من هذه المكونات بشكل تفصيلي ، ويتم ذلك أيضاً بتقسيم كل وظيفة رئيسية إلى الوظائف الفرعية المكونة لها ، ويستمر التقسيم على هذا النحو حتى تفكيك النظام بالكامل إلى وحدات فرعية (مكونات أو عمليات) بسيطة يسهل دراستها وفهمها .

3) **استخدام الطرق البيانية المختلفة Graphic methods** لتمثيل النظام ومكوناته وتبيان هيكليته بشكل تفصيلي . ومن الواضح أن استخدام هذه الرسوم تجعل بنية النظام أكثر وضوحاً مما يسهل فهمها ومعرفة تفاصيلها.

4) **استخدام التوثيق** لتسجيل جميع البيانات والمعلومات التي يتم جمعها وتحليلها ، وكذلك النتائج التي يتم التوصل إليها ، وتوثيق ذلك كله بأسلوب منظم يسهل عملية المراجعة والتدقيق قبل الانتقال إلى المرحلة التالية في عمليه التطوير .

5) **المراجعة والتدقيق** : يتضمن الأسلوب الهيكلي القيام بسلسلة من عمليات المراجعة والتدقيق للتأكد من جودة تنفيذ العمليات المطلوبة . فبعد إنجاز كل مرحله من مراحل التحليل والتصميم ينبغي إجراء مراجعة لمخرجات (وثائق) هذه المرحلة وتدقيقها للتأكد من مطابقتها للاحتياجات وخلوها من الأخطاء واكتمالها . وتتم هذه المراجعة أولاً مع أعضاء فريق التطوير ، ثم مع إدارة المنظمة .

7. وظيفة محلل ومصمم الأنظمة :

يقوم محلل الأنظمة عادة ببعض أو كل المهام و الانشطه الخاصه بتحليل النظام الحالي وتحديد متطلبات المستخدمين ثم تصميم النظام المناسب لتلبية هذه الإحتياجات. ويجب أن يتم ذلك من خلال التعاون الوثيق والفعال مع كل من الإدارة المسؤولة عن النظام والمستخدمين ، أي الموظفين الذين سيتعاملون مع النظام بشكل يومي .

في الماضي كان المحلل يقوم بجميع عمليات التحليل والتصميم ، إلا أن تعقد الأنظمة الحديثة وتطبيقاتها، وتنامي متطلبات الجودة، أدت إلى الاتجاه نحو فصل وظيفة التحليل عن التصميم. ويتزايد استخدام مصطلح مصمم الأنظمة كوظيفة مستقلة . وبشكل عام فإن عمليه تحليل وتصميم الأنظمة الحديثة لا يمكن أن تعتمد على جهد فردي بل هي مهمة جماعية يقوم بها فريق التطوير الذي يرأسه عاده خبير إستشاري في أنظمه المعلومات ويضم عدداً من المحللين والمصممين المختصين في تطبيقات مختلفة (نظم صناعيه – نظم ماليه ومحاسبيه – نظم تحكم آلي – نظم إحصائية وغيرها)، وفي مجالات محدده (قواعد البيانات – شبكات تراسل معطيات – أنظمة الوسائط المتعدده – أنظمة التعرف الآلي على الرموز والأصوات وهكذا) .

ونظراً لكون هذا الكتاب يدرس عملية تحليل الأنظمة وتصميمها فإننا سنقدم فيما يلي أهم الخصائص التي يجب أن تتوفر في المحلل أو المصمم الجيد وهي:

- حب العمل والمقدرة على التعامل الفعال مع الآخرين .
- المبادره والتعاون والدبلوماسية .
- القدره على تحفيز الآخرين للعمل والتعاون .
- القدرة على عرض الأفكار بطريقه جيده بشكل يقنع الآخرين بها.



- صعوبة اختيار أدوات وأساليب التحليل المناسبة حيث توجد عشرات الأدوات ، ولا توجد أداة واحدة تناسب تماماً جميع المشاريع، فلكل مشروع ظروفه وطبيعته وأفراده واحتياجاته ولذلك فإن الأدوات التي يجب استخدامها في التطوير مختلفة .
- ضرورة متابعة التطورات والإنجازات التقنية الحديثة في بيئة شديدة التغير ، وخاصة فيما يتعلق بتقنية الحواسيب والبرمجيات . وعلى المحلل أن يعرف جميع التفصيلات الفنية الدقيقة لكل منتج وكذلك أن يعرف لمن يتوجه للحصول على المعلومات المتعلقة بهذه المنتجات.
- تغير البيئة المحيطة بعالم الأعمال : من حيث المنتجات الجديدة والأساليب الجديدة و التشريعات الحكومية الجديدة وغيرها . ولذلك يجب أن يكون مُلماً بهذا المجال .
- إتقان فن التعامل مع الآخرين نظراً لأن المحلل يقضي معظم وقته مع الأفراد وليس مع التقنية ،ولذلك تتعلق معظم المشاكل التي يواجهها بعمل الأفراد وليس بالتقنية .

وأخيراً نورد فيما يلي مثالاً لبطاقة توصيف عمل محلل ومصمم الأنظمة توضح مهام ومتطلبات هذه الوظيفة الهامة .

بطاقة وصف وظيفة محلل ومصمم الأنظمة
وصف الوظيفة : تجميع وتحليل المعلومات المتعلقة بالنظام الحالي والمتطلبات المتعلقة بالأنظمة الجديـدة. يستخدم المحلل هذه المعلومات لتخطيط التعديلات في الأنظمة الحالية أو لتصميـم أنظمة جديده .ويقدم المحلل مواصفات التصميم مـن خـلال التقـارير الرسميـة والوثائـق، ويشرف على برمجة واختبار الأنظمة الجديدة وإعداد الموقع وعمليـات التوثيق والتدريب والتحويل والصيانة .
المهام : - تطبيق أساليب تجميع المعلومات لدراسة النظام الحالي وتحديد متطلبـات النظام الجديد المقترح . - تطوير حلول للمشاكل الموجودة في نظام العمل . - تصميم إجراءات تجميع ومعالجه البيانات . - استخدام المخططات الهيكليه وطرق التوثيق الأخرى لنمذجة الأنظمـة الحالية والجديدة . - وضع التقديرات المتعلقة بالوقت والموارد وكذلك فوائد النظام الجديد - القيام بتحليل التكلفة والمردود للحلول المقترحة . - الإشراف على إعداد الموقع واختيار التجهيزات والبرمجيات اللازمة . - بناء النماذج التجريبية للمساعدة في تطوير الأنظمة. - تقييم التصميمات من حيث الجودة وسهوله الصيانة . - تصميم نماذج المدخلات والتقارير والشاشات .

- تضمين إجراءات الحمايه ومتطلبات الأمن والسلامة في النظام . - الإشراف على البرمجة والاختبار والرقابة على جودتها . - الإشراف على إعداد وثائق الاستخدام وتدريب المستخدمين. - الإشراف على عملية التحويل إلى النظام الجديد . - الإشراف على عمليات الصيانة ومراقبه التغييرات . - تحديد مواصفات ومعايير قياسية لتطوير أنظمة المعلومات . - متابعة مستجدات تقنية الحواسيب والمعلومات .
الخبرات والمهارات : - بكالوريوس أو ماجستير في علوم الحاسب ، أو أنظمة المعلومات. - خبره في البرمجة وتحليل الأنظمة وتصميمها . - خبره في التدريب بشكل عام . - مهارات إتصالية و سلوكيه (الاتصال الشفهي، كتابه التقارير) . - خبرات إدارية .

أسئلة الفصل

1- عدد واشرح الخصائص العامة للأنظمة ؟

2- صنف الأنظمة بحسب درجة تعقيدها وطبيعتها ومصدرها ؟

3- عرف أنظمة المعلومات الحاسوبيه واشرح مكوناتها ؟

4- اشرح دور نظام المعلومات الحاسوبيه في النموذج العام للمنظمة؟

5- عدد واشرح المراحل الثلاث لتطوير أنظمة المعلومات الحاسوبيه والجهات المشاركة في عملية تطوير أنظمة المعلومات ؟

6- متى تظهر الحاجة إلى تحليل وتصميم أنظمة المعلومات ؟

7- ما هي أهداف تحليل وتصميم أنظمة المعلومات الحاسوبيه ؟

8- اشرح الأسلوب الهيكلي لتحليل وتصميم أنظمة المعلومات والمبادئ التي يقوم عليها ؟

9- أشرح أهم الخصائص التي يجب أن تتوفر في محلل الأنظمة والمهارات والمعارف التي يجب أن يمتلكها ؟

10- عدد أهم المهام التي تتضمنها وظيفة محلل الأنظمة ؟

الفصل الثاني

تطبيقات أنظمة المعلومات الحاسوبية في المنظمات

1. مقدمة

تهدف أنظمة المعلومات الحاسوبية إلى توفير المعلومات اللازمة لعمل الأفراد في المنظمات؛ أما طبيعة عمل هؤلاء الأفراد فهي على الأغلب إدارية، أي تتضمن مهام أساسية هي التخطيط والتنظيم والرقابة والتوجيه والتنسيق وغيرها. ويتم تنفيذ هذه المهام من خلال عمليات عديدة ومتنوعة تبدأ بالتسجيل والاحتساب والفهرسة وتخزين المعلومات ومعالجتها ثم استرجاعها وترتيبها بشكل يساعد على فهمها بصورة أسرع، وإيصالها إلى جميع الأشخاص والجهات التي يمكن أن تستفيد منها وتستخدمها في أنشطتها المختلفة. ويمكن أن نسمي هذه العمليات المذكورة أعلاه بالأعمال المكتبية Clerical works، وهي تعتبر من أبسط متطلبات العمل الإداري حيث تعتمد دقة وجودة العمل الإداري على دقة وجودة أداء هذه العمليات المكتبية البسيطة. وتعتبر أنظمة معالجة العمليات Transaction Processing Systems (TPS) وأنظمة أتمتة المكاتب Office Automation Systems (OAS) من أنظمة المعلومات الحاسوبية الموجهة لدعم هذا النوع من العمليات الإدارية.

أما النوع الآخر للعمليات الإدارية فهو تحليل المشكلات واتخاذ القرارات والذي يعتبر أساس وجوهر العمل الإداري. وإدارة المنظمة في جميع مستوياتها تقوم بهذا النوع من الأعمال ولكن بدرجات متفاوتة. ففي المستويات الدنيا يتم تحليل

المشكلات ذات الطبيعة البسيطة، والتي تسمى عادة بالمشكلات محددة البنية أو القابلة للبرمجة، واتخاذ القرارات المناسبة بشأنها.

أما المستويات الإدارية الوسطى فتتعامل مع المشكلات الأكثر تعقيداً، وهكذا حتى مستوى الإدارة العليا الذي يعالج عادة المشكلات بالغة التعقيد و التي تتعلق غالباً بالقضايا الاستراتيجية للمنظمة. وتوفر أنظمة المعلومات الحاسوبية تطبيقات عديدة لتلبية احتياجات الأفراد في جميع مستويات إدارة المنظمة.

وهكذا فإن أنظمة المعلومات الحاسوبية ترتبط إرتباطاً وثيقاً باحتياجات الأفراد والجماعات في مختلف مستويات إدارة المنظمة. ونظراً لتنوع هذه الاحتياجات، توجد اليوم عشرات، بل مئات الأنواع من الأنظمة الحاسوبية المستخدمة لتوفير المعلومات اللازمة لتلبية الاحتياجات الإدارية في مختلف المنظمات الاقتصادية والاجتماعية.

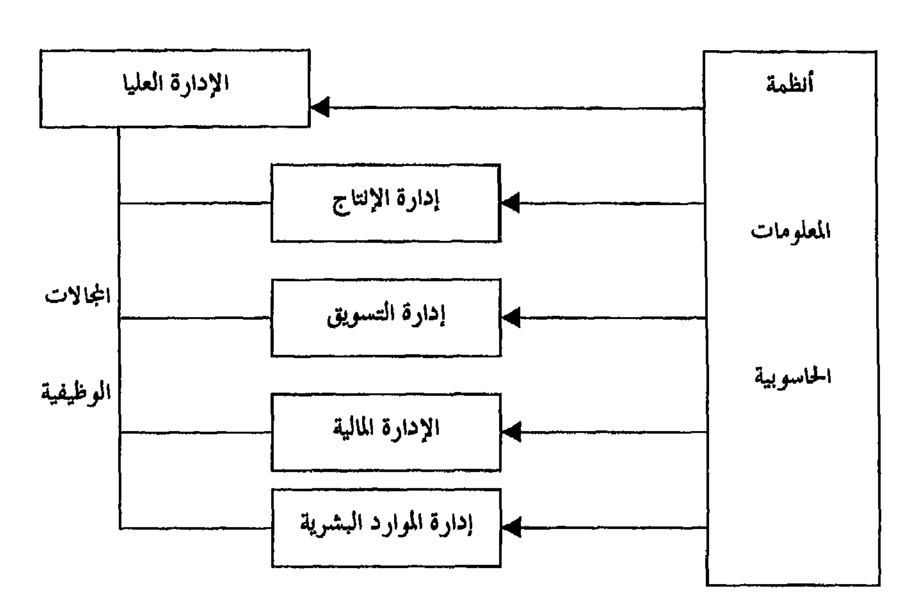
ويدرس هذا الفصل كيفية تصنيف التطبيقات المختلفة لأنظمة المعلومات الحاسوبية، ثم يقدم شرحاً لأهم أنواع التطبيقات بهدف التعريف بوظائفها ومكوناتها والعلاقات المتبادلة فيما بينها.

2- تصنيف أنظمة المعلومات الحاسوبية في المنظمة:

تهدف أنظمة المعلومات كما أشرنا فيما سبق إلى دعم وظائف المنظمة من خلال توفير المعلومات اللازمة لإدارة هذه الوظائف وتسييرها بالشكل الأمثل.

تتكون المنظمة عادة من مجموعة من الفعاليات الوظيفية Functional units التي تقوم كل منها بأداء جزء محدد من وظيفة المنظمة. فمثلاً يمكن اعتبار الشركات الصناعية مجموعة من الفعاليات الوظيفية مثل:

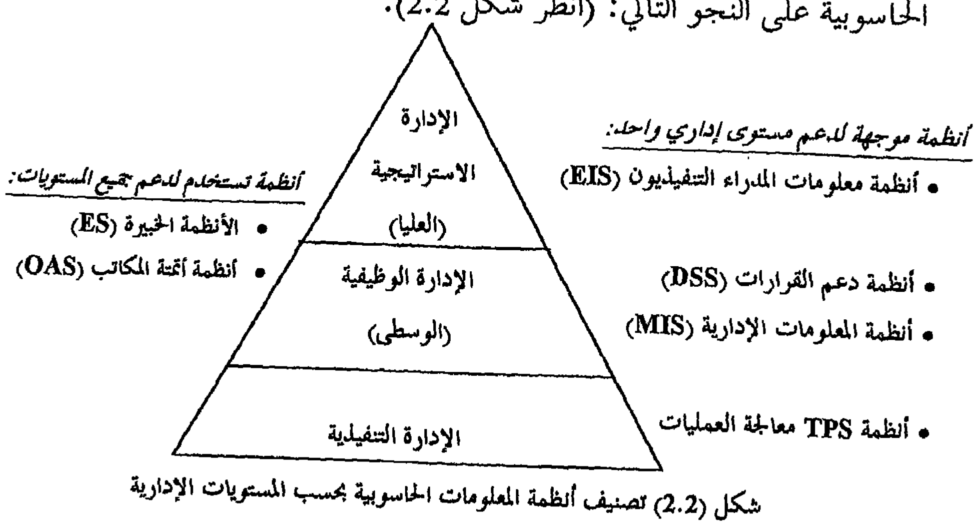
إدارات التسويق والإنتاج والمحاسبة والمشتريات والموارد البشرية وغيرهـــا. وكذلك الحال فان الجامعات يمكن اعتبارها مجموعة من الوحدات الوظيفية مثـــل: إدارات القبول والتسجيل والامتحانات والكليات والأقسام العلمية وغيرها. وتقـــوم أنظمة المعلومات بتوفير المعلومات اللازمة لمســاعدة هــذه الإدارات (الوحـــدات الوظيفية) في تحقيق أهدافها كما هو مبين في الشكل (1.2) أدناه:



شكل (1.2) أنظمة المعلومات الحاسوبية توفر المعلومات اللازمة للمنظمة

يمكن تصنيف أنظمة المعلومات الحاسوبية المستخدمة في المنظمـــات وفقـــاً لعدد من المعايير، وذلك على النحو التالي:

أ) بحسب المستوى الإداري الذي تدعمه يمكن تصنيف أنظمة المعلومات الحاسوبية على النحو التالي: (انظر شكل 2.2):



شكل (2.2) تصنيف أنظمة المعلومات الحاسوبية بحسب المستويات الإدارية

- **أنظمة معالجة العمليات (TPS) Transaction Processing Systems :**

وهي أنظمة معلومات موجهة لحوسبة العمليات التي يتم إنجازها في مستوى الإدارة التنفيذية (إدارة العمليات) في المنظمة.

- **أنظمة المعلومات الإدارية (Management Information Systems(MIS**

وهي أنظمة معلومات حاسوبية موجهة لدعم الإدارات الوظيفية في المنظمة.

وأهم هذه الأنظمة:

- نظام معلومات التسويق
- نظام معلومات الإنتاج
- نظام معلومات الموارد البشرية
- نظام المعلومات المالية

- وغيرها من أنظمة المعلومات الوظيفية.

- **أنظمة دعم القرارات Decision Support Systems (DSS):**

وهي أنظمة معلومات حاسوبية تقوم بتوفير الدعم اللازم لحل المشكلات المعقدة وذات البنية ضعيفة التحديد:

- **أنظمة معلومات المدراء التنفيذيون Executive Information(EIS) Systems:**

وهي أنظمة معلومات حاسوبية موجهة لدعم الإدارة العليا في المنظمة، وتركز على دعم القرارات المعقدة جدا وذات البنية شبه المحددة أو غير المحددة البنية.

- **أنظمة أتمتة المكاتب Office Automation Systems (OAS):**

وهي مجموعة من التطبيقات الحاسوبية المستخدمة لدعم الأعمال المكتبية من طباعة ورسم ومسك للسجلات وإعداد جداول وغيرها. وتوفر هذه الأنظمة بيئة اتصالات تفاعلية تدعم مجموعات العمل في مختلف مستويات المنظمة.

- **الأنظمة الخبيرة Expert Systems (ES):**

وهي أنظمة استشارية تستخدم لدعم عملية اتخاذ القرارات في مجالات معرفية محددة. وتستخدم في جميع المستويات الإدارية.

ب) بحسب حجم أنظمة المعلومات الحاسوبية يمكن تصنيفها إلى:

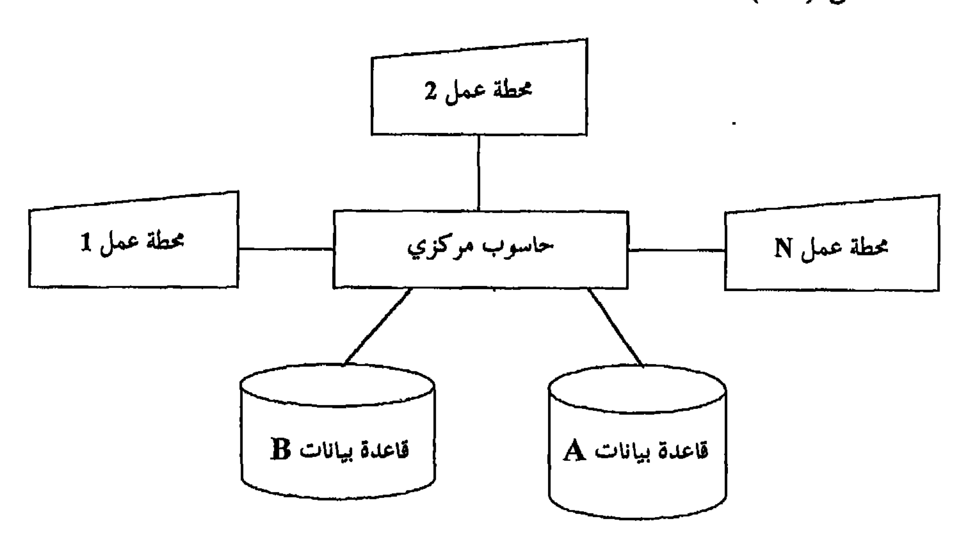
- **أنظمة معلومات حاسوبية لدعم عمل الأفراد:** توفر الدعم لمستخدم واحد أو عدد قليل جدا من المستخدمين، من خلال محطة عمل واحدة أو حاسب شخصي. وغالبا يتم تطوير مثل هذه الأنظمة من قبل المستخدم النهائي بنفسه، حيث يقوم باستخدام الأدوات البرمجية المعروفة كأنظمة إدارة قواعد البيانات أو الجداول الإلكترونية أو غيرها لبناء أنظمة محوسبة تلبي إحتياجاته الفردية.

وغالباً يقوم المستخدم بنفسه. أيضاً بإدخال البيانات واسترجاع المعلومات وطباعة التقارير اللازمة له.

- **أنظمة معلومات حاسوبية لدعم عمل المجموعات:**

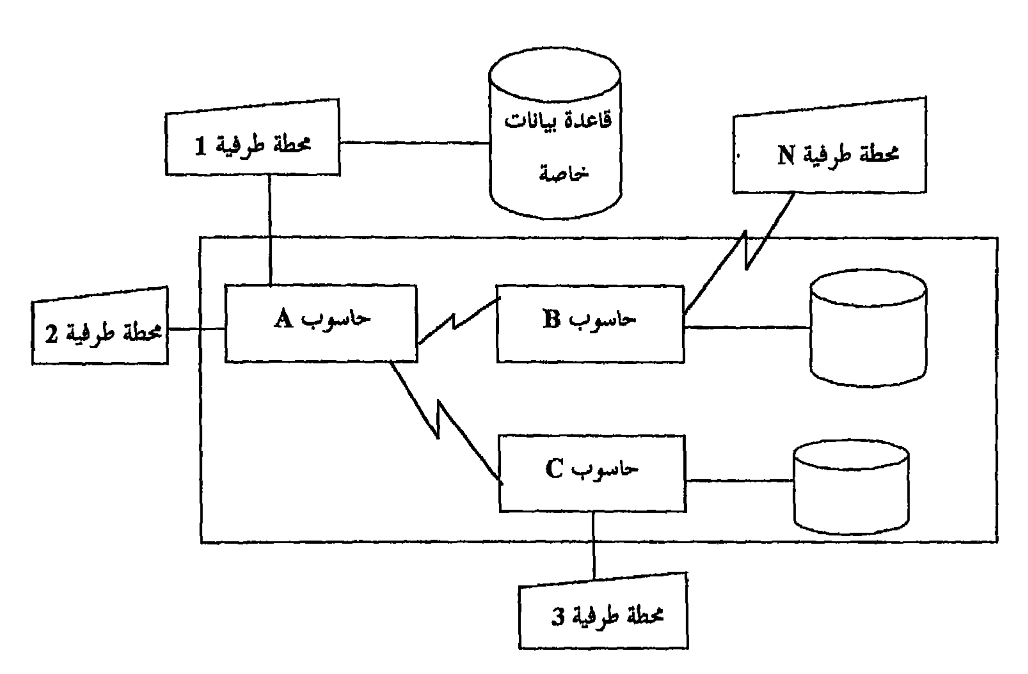
توفر هذه الأنظمة الدعم لعدد كبير من المستخدمين الذين ترتبط أعمالهم معاً في نظام حاسوبي واحد، أو باستخدام قواعد البيانات بشكل مشترك. ويمكن أن تبنى هذه الأنظمة بإحدى الطريقتين التاليتين:

- باستخدام حاسوب مركزي واحد، يتم فيه تخزين قواعد البيانات، والتعامل معها من خلال محطات طرفية تربط بهذا الحاسوب المركزي كما هو مبين في الشكل (3.2) أدناه:



شكل (3.2) نظام معلومات حاسبي متعدد المستخدمين (حاسوب مركزي)

- باستخدام أنظمة الشبكات الحاسوبية، حيث تربط المحطات الطرفية إلى عدة حواسب تتصل فيما بينها بشبكة اتصالات يتم من خلالها تبادل المعلومات بين هذه الحواسيب، كما هو مبين في الشكل (4.2) أدناه.



شكل (4.2) نظام معلومات حاسوبية متعدد المستخدمين (شبكة حواسب)

وبالطبع فان هذه الأنظمة متعددة المستخدمين التي تقوم عادة بدعم عمل المجموعات تكون أكثر تعقيداً من الأنظمة التي تدعم مستخدماً واحداً. ولذلك يجري تطوير هذه الأنظمة بإشراف إدارة أنظمة المعلومات في المنظمة حيث يتم تكوين فريق عمل ليقوم بعملية التطوير. ويضم هذا الفريق كما أشرنا في الفصل الأول محللين ومصممي أنظمة وإخصائيي قواعد بيانات وشبكات ومبرمجين وكذلك بعض المستخدمين الرئيسين للنظام. ويجب التركيز عند تطوير هذه الأنظمة على موضوعين رئيسيين هما توفير **الخصوصية** Privacy اللازمة لحماية معلومات الأفراد وعدم إنكشافها لغير الأشخاص المخولين باستخدامها. وكذلك

توفير إجراءات **الحماية والسلامة للنظام** Security لضمان عدم ضياع البيانات أو سرقتها أو إتلافها أو التلاعب فيها سواء عن قصد أو بغير قصد.

وأخيرا تجدر الإشارة إلى أن هذا النوع من الأنظمة الحاسوبية أصبحت تعتمد بشكل متزايد على استخدام الوسائط المتعددة لتأمين بيئة عمل أكثر كفاءة وفاعلية لمجموعات العمل المختلفة. حيث يتم من خلالها الإتصال بالكلمات المكتوبة والمنطوقة والصور والرسوم والأشكال البيانية والصور المتحركة وغيرها. مما يفتح آفاق واسعة أمام تطوير أنظمة المعلومات الحاسوبية في المستقبل.

3– أنظمة معالجة العمليات Transaction Processing Systems (TPS)

هي أنظمة معلومات حاسوبية تقوم بمعالجة البيانات المتعلقة بعمليات المنظمة، وذلك من خلال تجميعها وتدقيقها وتخزينها ثم تحويلها إلى معلومات، وإتاحة هذه المعلومات للمستفيدين منها داخل المنظمة وخارجها، وتتعامل هذه الأنظمة بشكل مباشر مع قاعدة البيانات حيث تقوم بإدخال البيانات إليها وتحديثها وكذلك استرجاعها وإخراجها بشكل استعلامات وتقارير لتلبية الاحتياجات المختلفة للمستخدمين.

وبشكل عام تقوم أنظمة معالجة العمليات هذه TPS بالمهام الرئيسية التالية:

1) **تجميع البيانات**: عن العمليات الداخلية والخارجية ذات العلاقة بعمل المنظمة (شراء – بيع – تخزين – توظيف – صيانة – إنتاج...) وهذا يتضمن أيضا التأكد من صحة هذه البيانات ودقتها واكتمالها وانسجامها مع البيانات الموجودة قبل إدخالها إلى قاعدة البيانات.

2) **تداول البيانات Manipulation**: وهذا يشمل العمليات التالية:

- التصنيف Classification.

- الفرز Sorting.
- الاحتسابCalculation .
- التلخيص Summarizing.
- وغيرها من عمليات المعالجة.

3) **تخزين البيانات** في قاعدة البيانات التي تضم عادة بيانات تفصيلية عن جميع العمليات التي تحدث في النظام.

4) **استرجاع المعلومات:** وهذا يعني إنتاج المخرجات المختلفة من تقارير واستعلامات لتلبية احتياجات الأفراد والمجموعات داخل المنظمة وخارجها. وتتم عمليات الاسترجاع هذه بشكل دوري، أو حسب الطلب عند ظهور الحاجة إلى تقرير أو استعلام معين.

وأخيرا نوجز فيما يلي أهم **خصائص أنظمة معالجة العمليات** على النحو التالي:

- **موجهة لحوسبة أداء العمليات الأساسية** في المنظمة كالمحاسبة وإعداد الفواتير ورقابة حركة المخزون وإصدار أوامر الشراء واستلام المواد ومعالجة طلبات الزبائن وغيرها.
- **تتم وفق إجراءات نمطية قياسية موحدة** تقريبا في جميع المنظمات. وذلك نظرا لأن جميع المنظمات تستخدم إجراءات متشابهة لتنفيذ العمليات المتعلقة بالمحاسبة وشؤون الموظفين وإعداد الموازنات ومعالجة طلبات الزبائن وغيرها.
- **تتعامل مع البيانات التفصيلية** المتعلقة بأنشطة ومهام المنظمة، وتوفر تقارير تفصيلية عنها لأغراض التدقيق والمراجعة.
- تتعامل بشكل رئيسي مع المعلومات المتعلقة **بالأحداث الماضية (معلومات تاريخية)**، ولا توفر معلومات حول التوقعات المستقبلية.

- **توفر إمكانات محدودة لتحليل المشكلات واتخاذ القرارات** وذلك نظرا لتركيزها على حوسبة العمليات وتجميع بياناتها التفصيلية.

وأخيرا تجدر الإشارة إلى أن معالجة العمليات في هذه الأنظمة يمكن أن تتم وفق طريقة المعالجة المباشرة On-Line Processing، أو وفق طريقة معالجة الدفعات Batch Processing. وبالرغم من أن المعالجة المباشرة تعتبر طريقة أكثر تطورا وكفاءة وفاعلية ولكن بعض أنظمة معالجة العمليات ذات الطبيعة الدورية، كإعداد كشوف الرواتب والأجور وكشوف الترقيات السنوية وجرد المخزون وغيرها، يتم معالجتها بطريقة الدفعات نظرا لطبيعة عملياتها التسلسلية.

4) أنظمة دعم القرارات Decision Support Systems (DSS) :

هي أنظمة معلومات حاسوبية موجهة لمساعدة المستخدم في تحليل المشكلات المعقدة والوصول إلى القرارات اللازمة لحلها. ونظرا للتنوع الكبير في المشاكل فانه تستخدم طرق عديدة لتحليل المشكلات واتخاذ القرارات. فبعض المشاكل يتطلب استخدام خوارزميات رياضية للوصول إلى الحل الأمثل كما هو الحال في مسائل البرمجة الخطية بشكل عام. بينما تتطلب مشاكل أخرى القيام بتجارب واختبارات عديدة يتم خلالها دراسة تأثير المدخلات المختلفة (تجريب البدائل المختلفة) للوصول إلى الحل المناسب. وفي معظم أنظمة دعم القرارات يقوم المستخدم في البداية بصياغة النموذج الرياضي المناسب للمشكلة التي تجري دراستها، ثم يقوم بتطوير هذا النموذج حتى يصبح معبرا بالدقة المطلوبة عن هذه المشكلة. وأخيرا يقوم بإدخال البيانات إلى النموذج وحله واستخراج النتائج. وبشكل عام تتميز أنظمة دعم القرارات بالخصائص التالية:

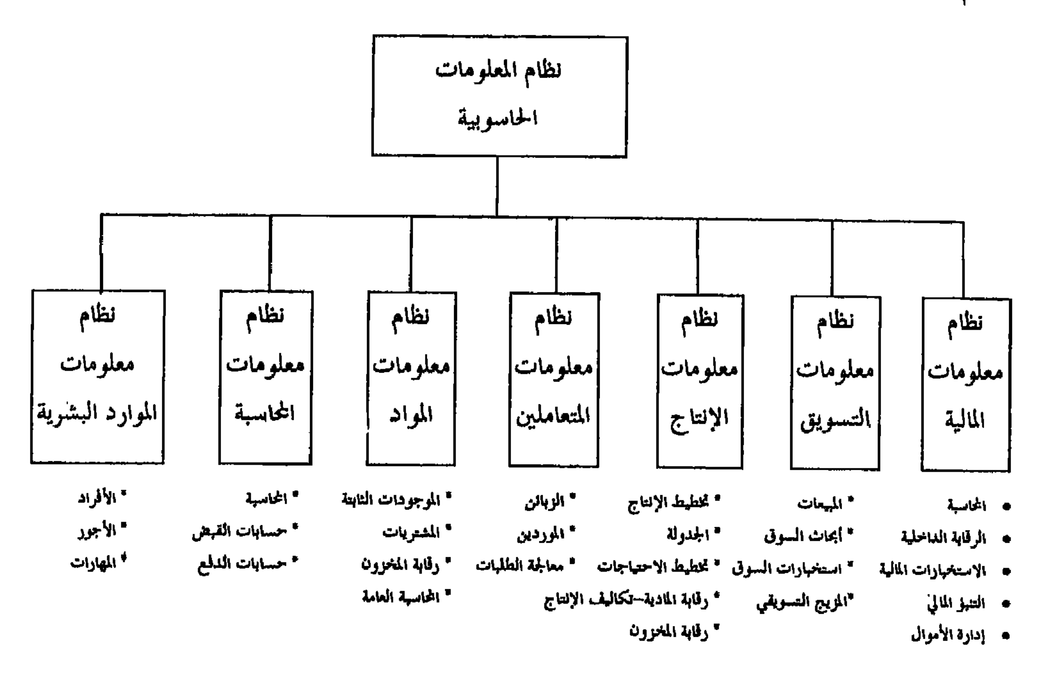
- تعتبر أنظمة معلومات حاسوبية **ذات طبيعة تفاعلية Interactive** يقوم المستخدم من خلالها بإدخال الفرضيات وتجريب الاستراتيجيات المختلفة لمعرفة تأثيراتها المختلفة، حتى الوصول إلى النتائج المطلوبة.
- تعتبر أنظمة دعم القرارات **أداة جيدة لحل المشكلات المعقدة** ذات البنية شبه المحددة Semi Structured أو غير المحددة Unstructured. ولذلك فإن **استخدام النماذج الرياضية** يعتبر مكونا أساسيا من مكونات هذه الأنظمة.
- تتطلب هذه الأنظمة **استخدام بيانات كثيرة ومن مصادر متعددة**. ولذلك فان قواعد البيانات بمختلف أنواعها يمثل أيضا مكونا أساسيا هاما من مكونات هذه الأنظمة.
- تصمم أنظمة مساندة القرارات **للمساعدة في حل المشكلات** ولذلك تسمى Problem- Oriented. فهي تركز على توفير الوسائل والأدوات والبيانات اللازمة لتحليل المشكلات. وهذا ما يميزها عن أنظمة المعلومات الأخرى الموجهة عادة نحو عمليات المعالجة Process-Oriented كما هو الحال في أنظمة المعلومات الوظيفية وأنظمة معالجة العمليات.

5) نماذج من أنظمة المعلومات الحاسوبية في المنظمة.

يتبين من خلال العرض السابق ضرورة وجود العديد من أنظمة المعلومات الحاسوبية في المنظمة. حيث يقوم كل منها بتوفير المعلومات اللازمة لمستوى إداري معين وضمن مجال محدد. وهذا لا يعني أن هذه الأنظمة مستقلة ومنفصلة عن بعضها البعض. بل تكون غالبا متكاملة بحسب الوظائف والمجالات التي تقوم بدعمها، كما تتكامل أيضا في إطار المنظمة ككل. وغالبا يتجسد هذا التكامل من خلال استخدام

قاعدة بيانات موحدة تتشارك في استخدامها هذه الأنظمة، وكذلك من خلال تدفقات البيانات التي تربط الأنظمة ذات العلاقة ببعضها البعض.

وسنتعرف فيما يلي على نماذج مختارة من أنظمة المعلومات الحاسوبية التي يمكن أن نجدها في المنظمات المختلفة وعلاقة هذه الأنظمة بعضها مع بعض. ويبين الشكل (5.2) مخططا هيكليا لأنظمة المعلومات الحاسوبية التي يمكن أن نجدها في معظم منظمات الأعمال.



شكل (5.2) هيكلية أنظمة المعلومات الحاسوبية في منظمات الأعمال

1.5 نظام معلومات الموارد البشرية

يهدف هذا النظام إلى دعم وظيفة الموارد البشرية في المنظمة، ويقوم بتخزين ومعالجة البيانات التفصيلية المتعلقة بالموارد البشرية بهدف تزويد إدارة المنظمة

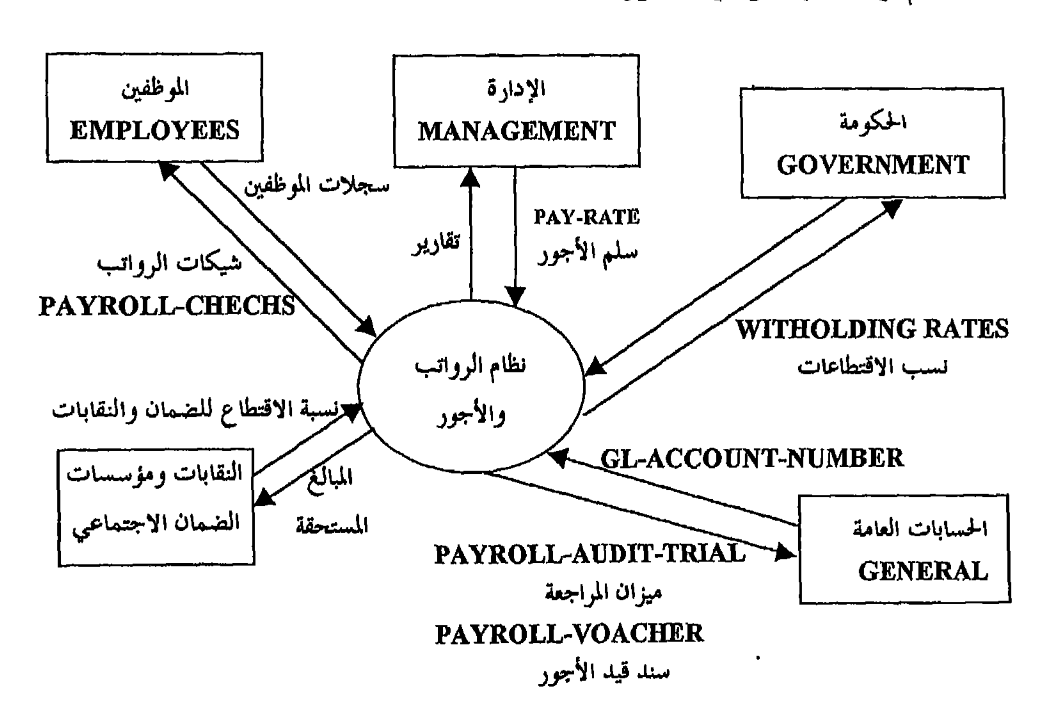
بالمعلومات اللازمة لتحقيق الاستخدام الأمثل لمواردها البشرية. وهذا يشمل وظـائف تخطيط القوى العاملة وتنظيم عمليات التوظيف والتدريـــب والتـأهيل والتحفـيز وغيرها. يمكن أن يتضمن نظام إدارة الموارد البشرية عدة أنظمة فرعية أهمها:

أ) نظام سجلات الأفراد Personnel Records Subsystem.

ب) نظام الرواتب والأجور Payroll Subsystem.

جـ) بنك المهارات الفنية والإدارية Skills Inventory .

وفيما يلي المخطط البيئي لنظام الرواتب والأجور الذي يبين مدخـــلات هـذا النظام ومصادرها ومخرجاته ووجهاتها المختلفة.



شكل (6.2) المخطط البيئي لنظام الرواتب والأجور

2.5 نظام المعلومات المحاسبية Accounting IS

يقوم هذا النظام بتسجيل وتخزين جميع البيانات المتعلقة بالعمليات المالية في المنظمة وهو يتألف بدوره من ثلاث أنظمة فرعية هي:

أ) نظام حسابات القبض Accounts Receivable :

ويسمى هذا النظام أيضا نظام حسابات الزبائن، ويهتم بمتابعة تحصيل أموال المنظمة الموجودة لدى الغير. حيث يقوم بإعداد وإصدار الفواتير الخاصة بالمبيعات وتدقيق حدود الإئتمان المسموح بها للزبائن، وكذلك تسجيل الدفعات التي تصل من الزبائن، وإرسال مذكرات المطالبة لهم في حال تأخرهم عن الدفع.

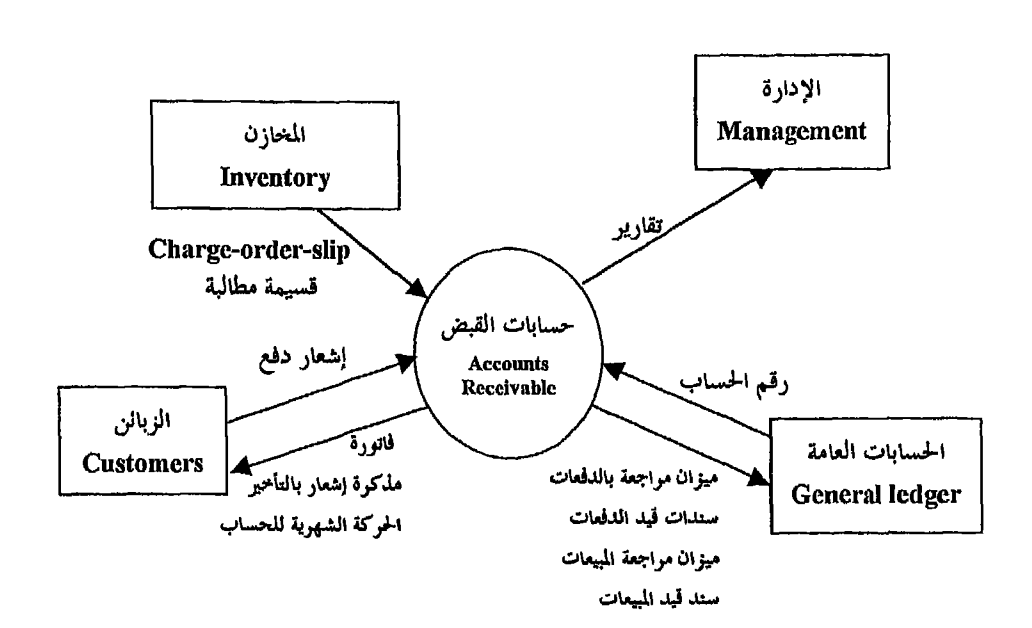
يقوم هذا النظام بمسك سجل خاص لكل زبون من زبائن المنظمة يسجل فيه قيم الفواتير المطلوبة منهم والدفعات التي ترد للمنظمة، وتدقيق هذه الدفعات مع الفواتير، كما يمكن أن تقوم بعض هذه الأنظمة بإعطاء حسومات للدفع المبكر، ولذلك يقوم بتحديد تاريخ الدفعة لمنح الحسومات المناسبة بشأنها.

ويقوم هذا النظام بتحديد الزبائن الذين يتأخرون بشكل دائم في تسديد المبالغ المطلوبة منهم لاتخاذ الإجراءات المناسبة بشأنهم.

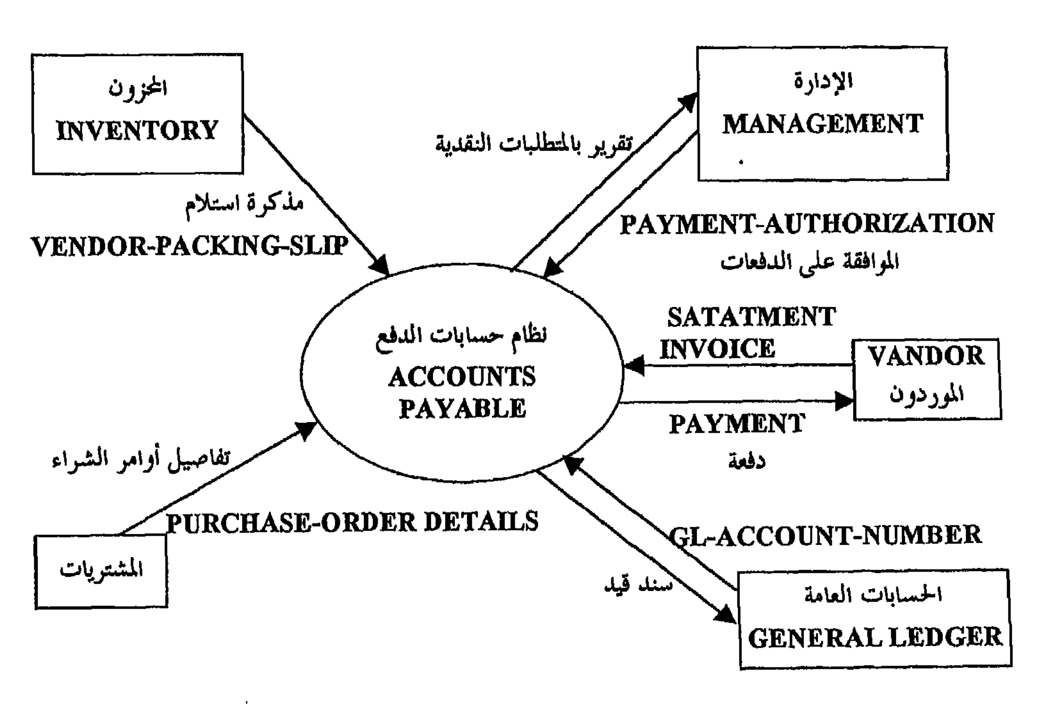
كما يقوم هذا النظام بإصدار تقارير دورية تبين المبالغ التي تم تحصيلها والمبالغ المتبقية لدى الزبائن مرتبة حسب فتراتها الزمنية Aging Report. ويبين المخطط التالي مدخلات هذا النظام ومصادرها وكذلك مخرجاته أو وجهته.

ب) نظام حسابات الدفع Accounts Payable:

ويسمى هذا النظام الفرعي أيضا نظام حسابات الموردين، ويقوم هذا النظام بمتابعة عمليات دفع الأموال المستحقة على المنظمة، حيث يقوم بمتابعة عمليات الشراء وتسجيل أوامر الشراء المرسلة إلى الموردين، ثم استلام الفواتير



شكل (7.2) المخطط البيئي لنظام حسابات القبض

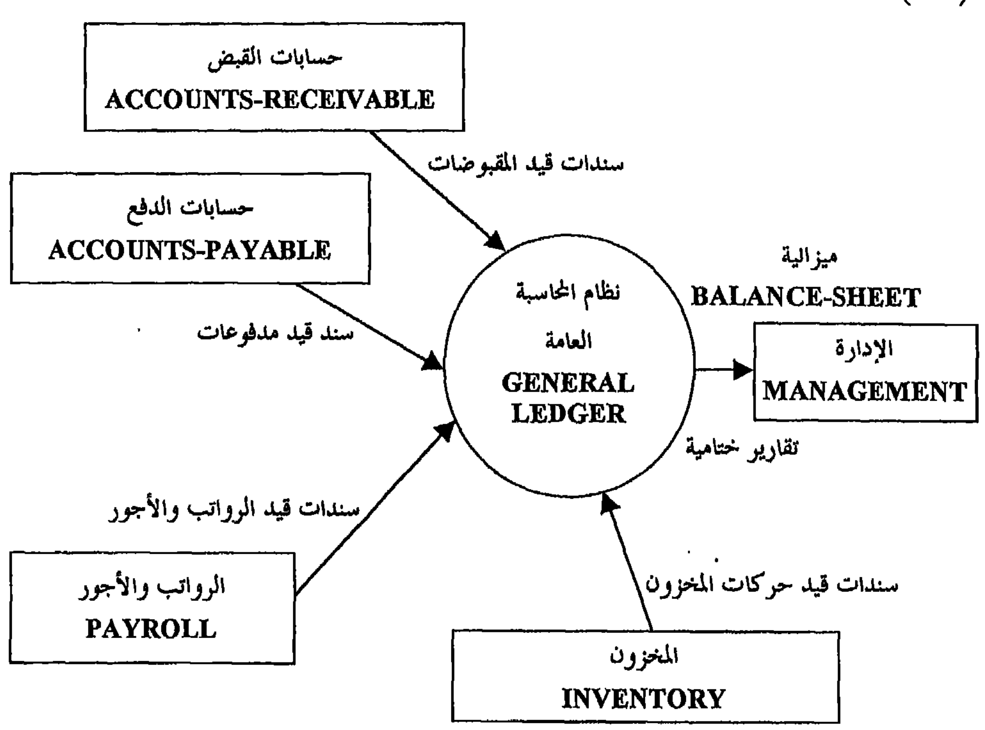


شكل (8.2) المخطط البيئي لنظام حسابات الدفع (محاسبة الموردين)

المرسلة منهم وتدقيقها مع أوامر الشراء ومذكـرات الاسـتلام وإصـدار الشيكات وإرسالها إلى الموردين، ويبين الشكل (8.2) المخطط البيئي لهذا النظـام الفرعي.

جـ) نظام الحسابات العامة General Accounts :

يقوم هذا النظام بمتابعة موجودات المنظمة وأرصدتها وإصـدار التقـارير المختلفة الخاصة بالحالة المالية العامة كالميزانية وحساب الأرباح والخسائر وغيرهـا. ويقوم هذا النظام بتجميع البيانات من عدة أنظمة فرعية ثم يسـتخدمها لإصـدار الموازنة والتقارير الختامية المختلفة كما هو مبين في المخطـط البيئـي في الشـكل (9.2).



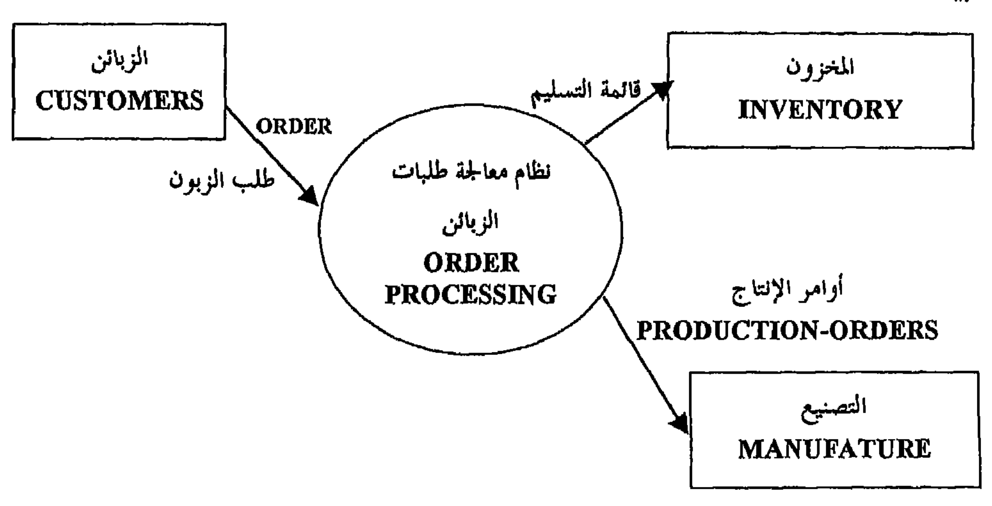
شكل (9.2) المخطط البيئي لنظام المحاسبة العامة

3.5 نظام معلومات التسويق Marketing Information Systems:

يهدف هذا النظام إلى دعم وظيفة التسويق في المنظمة من خلال توفير المعلومات اللازمة لها من داخل و خارج المنظمة. فالتسويق يهتم بشكل عام بالتعريف بالمنتجات والخدمات التي توفرها المنظمة. ويجب أن يشمل هذا التعريف ليس فقط الزبائن الحاليين، وإنما أيضا الزبائن المحتملين الذين يمكن أن يستفيدوا من هذه المنتجات أو الخدمات في المستقبل. ويتألف نظام معلومات التسويق من الأنظمة الفرعية التالية:

أ. نظام المبيعات Sales Subsystem:

ويقوم بتنفيذ عمليات البيع وتجميع ومعالجة عمليات المبيعات وتخزينها في قاعدة البيانات. ويسمى هذا النظام أيضا نظام معالجة الطلبات أو نظام إدخال الطلبات Order Processing System ويبين الشكل (10.2) المخطط البيئي لنظام المبيعات.



شكل (10.2) المخطط البيئي لنظام معالجة الطلبات

ب. نظام أبحاث التسويق Marketing Research Subsystem :

ويهتم بتوفير جميع البيانات المتعلقة بزبائن المنظمة الحاليين والمستقبليين وتحديد احتياجاتهم واتجاهات تطورها. يحصل هذا النظام على مدخلاته من البيانات التي تم تجميعها من خلال نظام المبيعات والمسوحات والدراسات والإستقصاءات التي يتم تنظيمها من قبل إدارة التسويق لهذا الغرض.

جـ) نظام استخبارات السوق Marketing Intelligence Subsystem

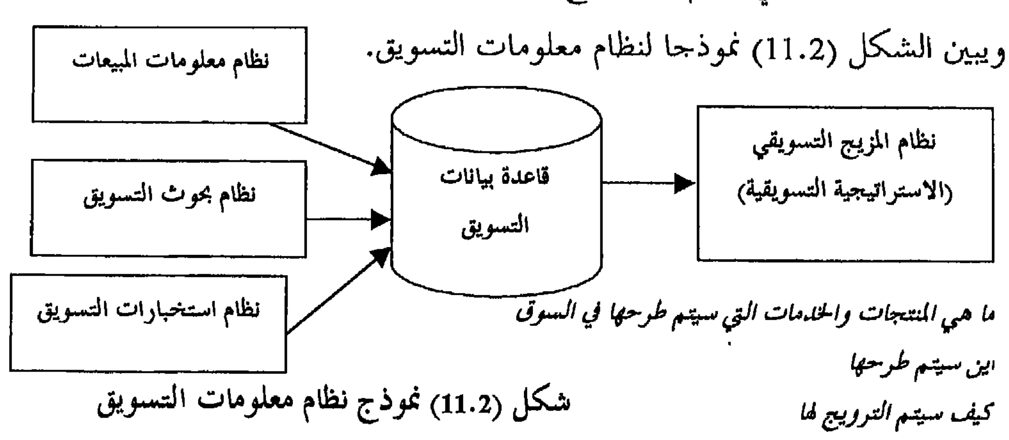
يهتم هذا النظام بتوفير البيانات المتعلقة بالمنافسين للتعرف على أوضاعهم الحالية وكذلك على خططهم المستقبلية واستراتيجياتهم التسويقية.

د) نظام المزيج التسويقي Marketing Mix Subsystem

يستخدم هذا النظام مخرجات الأنظمة الفرعية الثلاث السابقة لتحديد المزيج التسويقي الأمثل للمنظمة والذي يتضمن تحديد:

- المنتجات والخدمات المطلوب تسويقها.
- الأسواق التي سيتم فيها طرح هذه المنتجات والخدمات.
- أساليب الترويج التي سيتم استخدامها لذلك.
- الأسعار التي سيتم وفقها بيع هذه المنتجات والخدمات.

ويبين الشكل (11.2) نموذجا لنظام معلومات التسويق.



شكل (11.2) نموذج نظام معلومات التسويق

4.5 نظام معلومات التصنيع Manufacturing IS

بالإضافة إلى استخدام الحاسوب في عمليات التصميم Computer-Aided Design (CAD) والتصنيع Computer-Aided Manufacturing (CAM)، فهو يستخدم أيضا في إدارة العمليات الإنتاجية كجدولة الإنتاج ورقابة المخزون وضبط الجودة ومتابعة تكاليف الإنتاج. تمثل هذه المجموعة الثانية من التطبيقات ما يطلق عليه نظام معلومات التصنيع.أو نظام معلومات إدارة الإنتاج Production Management IS. يعتبر نظام معلومات إدارة الإنتاج من الأنظمة الرئيسية في الشركات الصناعية، ويقوم عادة بتجميع المعلومات المتعلقة بالتسهيلات الإنتاجية كالآلات والمعدات والمواد والمهارات (الموارد البشرية) وجدولة استخدامها لإنتاج المنتجات المختلفة.

وهكذا فان نظام إدارة الإنتاج يتكون من الأنظمة الفرعية التالية:

- **النظام الفرعي لتخطيط الإنتاج** Planning Subsystem الذي يقوم بتحديد خطة الإنتاج (المنتجات المطلوب تصنيعها وكمياتها والموارد اللازمة لها).
- **النظام الفرعي لجدولة الإنتاج** Scheduling Subsystem الذي يقوم بعمليات تخصيص موارد الإنتاج وبشكل خاص الآلات والمعدات لإنتاج الكميات المطلوبة (لتنفيذ خطة الإنتاج) ويسمى هذا النظام الفرعي أيضا نظام تخطيط الطاقات الإنتاجية Capacity Planning.
- **نظام تخطيط الاحتياجات المادية** Materials Requirement Planning (MRP) الذي يقوم بتحديد المستلزمات المادية لخطة الإنتاج ووضع خطة إصدار أوامر الشراء اللازمة لتأمين هذه المواد. ويتضمن هذا النظام نظاما فرعيا هاما يعتبر نواة عمل هذا النظام وهو نظام بنية المنتجات Product

Structure أو قائمة المواد Bill of Materials (BOM) الذي يقوم بتخزين المعلومات المتعلقة بالبنية الهرمية لجميع المنتجات التي يتم تصنيعها في الشركة.

- **النظام الفرعي لتكاليف الإنتاج** Production Cost Subsystem الذي يقوم بمتابعة التكاليف الفعلية لعمليات الإنتاج ومقارنتها مع التكاليف المعيارية أو المخططة بهدف التحكم الفعال فيها.
- **النظام الفرعي لرقابة المخزون** Inventory Control Subsystem: ويقوم برقابة مخزون المواد الأولية وقطع الغيار اللازمة لاستمرارية عمليات التصنيع وتنظيم صرفها و إعادة طلبها بحيث يتم بشكل دائم تلبية احتياجات التصنيع وبأقل تكلفة ممكنة وسيتم التعرف على هذا النظام الفرعي في الفقرة التالية

5.5 نظام معلومات إدارة المواد Materials Management Information System

يشمل المفهوم الحديث لإدارة المواد حركة هذه المواد داخل المنظمة بدأ من عمليات الشراء والاستلام مرورا بعمليات التخزين والتصنيع وانتهاء بتسليم المنتجات إلى الزبائن.

ووفقا لذلك فان عمليات التخزين والتصنيع التي تعرفنا عليها في الفقرة السابقة تقع ضمن هذا المفهوم الشامل لإدارة المواد. وهذه هي طبيعة الحال في الشركات الصناعية، حيث يتم تخطيط المشتريات من الموردين في ضوء توقعات المبيعات أو في ضوء الطلبات التي يقدمها الزبائن للحصول على منتجات أو خدمات الشركة. أما في الشركات التجارية فان مفهوم إدارة المواد ينحصر في عمليات الشراء والتخزين للمواد والمنتجات التي تتعامل بها هذه الشركات. وبذلك يكون نظام رقابة المخزون هو النواة الأساسية لإدارة المواد في الشركات التجارية.

أما أهم وظائف هذا النظام فهي:

- تسليم المواد والقطع وغيرها من موجودات المخزون إلى الأقسام الإنتاجية.
- الاستعلام أي توفير المعلومات المتعلقة بفقرات المخزون وأرصدتها الكمية والقيمية وحركاتها ومصادر الحصول عليها.
- إعادة طلب المواد التي يصل رصيدها إلى مستوى إعادة الطلب.
- جرد المخزون ويقصد به التأكد من تطابق الموجودات الفعلية للمخزون مع البيانات المدونة في السجلات.
- استلام المواد والقطع وتسجيل بيانات هذه العمليات في سجلات المخزون.

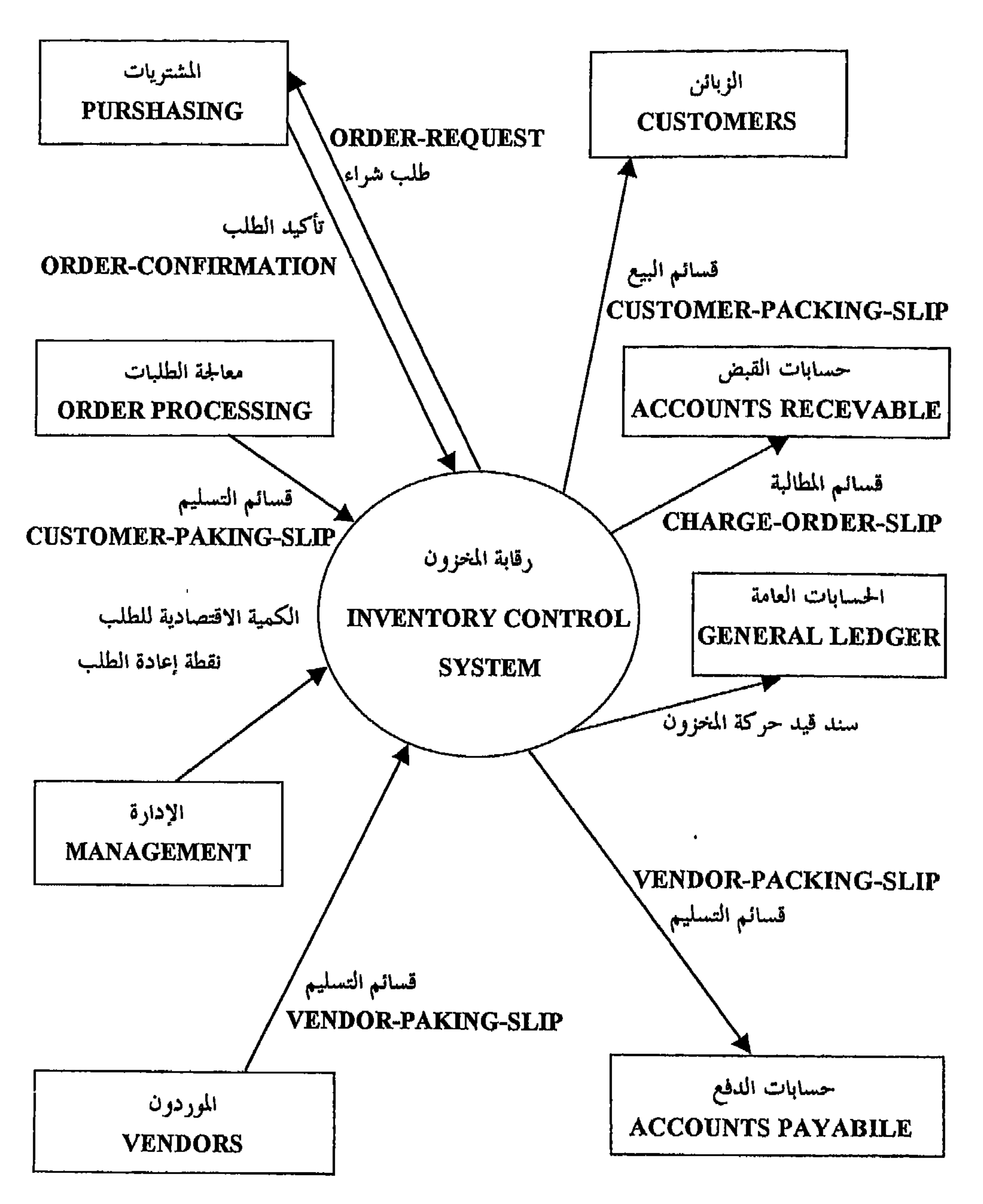
ويبين الشكل (12.2) المخطط البيئي لنظام رقابة المخزون، حيث يتضح الموقع المركزي لهذا النظام الذي يرتبط تقريبا مع العديد من أنظمة المعلومات الموجودة في المنظمة.

6.5 نظام المعلومات المالية Financial Information System

يهتم هذا النظام بتوفير المعلومات المتعلقة بالتدفقات النقدية Money flow من وإلى المنظمة، ويساعد المدراء في إدارة الاستثمارات وتوفير السيولة اللازمة لإدارة موارد المنظمة. وبشكل عام يتألف هذا النظام من الأنظمة الفرعية التالية:

- **نظام المعلومات المحاسبية** الذي تمت دراسته في فقرة سابقة من هذا الفصل.
- **نظام المراجعة الداخلية** Internal Audit Subsystem:

يقوم هذا النظام بتدقيق السجلات المحاسبية للتأكد من دقتها وصحتها وتقييم عمليات الشركة من وجهة النظر المالية.



شكل (12.2) المخطط البيئي لنظام رقابة المخزون

نظام الاستخبارات المالية Financial Intelligence Subsystem

ويقوم بتجميع المعلومات المتعلقة بعناصر البيئة التي تعمل ضمنها المنظمة والتي لها تأثير مباشر أو غير مباشر على تدفق الأموال من وإلى المنظمة. وهذا يشمل المؤسسات المالية وحملة الأسهم والمالكين والجهات الحكومية ذات العلاقة.

- **نظام التنبؤات المالية** Forecasting subsystem: ويتم من خلاله الحصول على توقعات بعيدة المدى تساعد في التخطيط الاستراتيجي للمنظمة.
- **نظام إدارة الأرصدة المالية** Funds Management الذي يهتم بإدارة تدفقات الأموال من وإلى المنظمة.
- **نظام الرقابة المالية** ويقوم بإعداد ميزانية التشغيل وتوفير معلومات التغذية العكسية (الراجعة) للمدراء لتمكينهم من ضبط النفقات الفعلية ومقارنتها مع الميزانية المخصصة.

6. العمليات الإدارية في المنظمة.

يتضح من العرض السابق لأنظمة المعلومات الحاسوبية أن هذه الأنظمة لا توجد بشكل مستقل عن بعضها البعض، بل تتفاعل معا من خلال تدفق البيانات فيما بينها. فمثلا البيانات المتعلقة بطلبات الزبائن أو أوامر الشراء ترسل إلى الأنظمة المحاسبية لإنجاز عمليات إعداد الفواتير أو التدقيق والدفع، وبدورها ترسل هذه البيانات إلى نظام المحاسبة العامة لإجراء القيود اللازمة في الحسابات الخاصة بهذه العمليات. وبشكل عام فإنه في ضوء الهدف الرئيسي لهذه الأنظمة الحاسوبية وهو توفير المعلومات اللازمة للعمليات الإدارية Business Processes في المنظمة، فإنه يصبح من الأهمية بمكان التعرف على هذه العمليات بهدف فهم طبيعتها ومتطلباتها وصولا إلى تحديد أهداف أنظمة المعلومات الحاسوبية اللازمة لدعمها.

تعرف العملية الإدارية بأنها مجموعة الخطوات اللازمة لتحقيق هدف إداري محدد. وبذلك فان العمليات الإدارية تحدد الطريقة التي يتم بها إنجاز الأعمال المختلفة في المنظمة، وهذا يشمل الطرق المطلوب اتباعها لتحقيق الأهداف، وكيفية التفاعل بين الأطراف المشاركة بها.

وبشكل عام تقوم المنظمات بتنفيذ مجموعة كبيرة من العمليات الإدارية المختلفة. ويمكن تصنيف هذه العمليات بطرق مختلفة. ويتم التركيز عادة على العمليات الحرجة Critical Processes التي تعتبر أساسية وهامة لبقاء المنظمة واستمرار نجاحها. وتكون هذه العمليات غالبا مرتبطة بتقديم المنتجات والخدمات لزبائن المنظمة لتلبية احتياجات محددة لديهم. ويتم تحديد العمليات الإدارية في المنظمة من خلال بنية هرمية في عدة مستويات بحيث تؤدي العمليات الموجودة في مستوى أعلى إلى تنشيط عمليات في المستويات التابعة لها (الأدنى). فمثلا عملية تلبية طلب الزبون تستدعي عدة عمليات ثانوية مثل إعداد الفاتورة وإرسالها إلى الزبون (في المحاسبة)، وعملية طلب مواد من المستودع، وهكذا.

وعند دراسة العمليات الإدارية في المنظمة يجب الاهتمام بمتطلبات الجودة اللازمة للتأكد من تنفيذ هذه العمليات وفقا لهذه المتطلبات للتأكد من جودتها. ويقصد بجودة العملية خلوها من الأخطاء وتقليل الأحداث غير المتوقعة، وتحقيق أهداف العملية بأقل استخدام للموارد (أقل التكاليف)، وأن يكون كل من الزبائن والعاملين في المنظمة راضين عن إنجازها.

وللتعرف على طبيعة العمليات الإدارية يمكن تصنيفها وفقا لدورها وأهميتها إلى عمليات رئيسية وعمليات مساندة. تتضمن العمليات الرئيسة جميع العمليات التي تمثل أساس عمل المنظمة. فمثلا في الشركات الصناعية تكون عمليات الإنتاج

والتسويق (البيع) وتخطيط الاحتياجات المادية ورقابة المخزون والمشتريات والصيانة وإدارة الجودة بمثابة عمليات رئيسة، أما العمليات المساندة فتتمثـــل في البحـــوث والتطوير والتخطيط الاستراتيجي والإدارة المالية والمحاسبة و إدارة الموارد البشـــرية وغيرها من العمليات التي تضمن توفير البيئة المناسبة للعمليات الرئيسة.

فمثلا تضمن العمليات المحاسبية توفير الأموال اللازمة لشراء الاحتياجـــات المادية وتنفيذ عمليات الإنتاج والتسويق وهكذا. كما تعتبر عمليات إدارة المـــوارد البشرية من تعيين وتدريب وتحفيز وغيرها ضمان لتوفير المهارات اللازمـــة للقيـــام بالعمليات الرئيسية من تسويق و انتاج ورقابة جودة وغيرها.

إن دراسة العمليات في المنظمة، والتعرف علـــى خصائصـــها ومتطلبـــات تنفيذهـــا يساعد في تحديد أنظمة المعلومات الحاسوبية اللازمـــة لحوســـبة هـــذه العمليات. كما تساعد هذه الدراسة في تحديد أولويات تطوير هذه الأنظمة، حيـــث يجب التركيز في المقام الأول على مساندة العمليات الرئيسية ثم المساندة. وفي النهاية فإن هذا التحليل يساعد المنظمة في تكوين صورة شاملة عن أنظمـــة المعلومـــات اللازمة لها، وأولويات تنفيذها والموارد اللازمة لذلك.

أسئلة الفصل:

1. كيف يمكن تصنيف أنظمة المعلومات الحاسوبية؟
2. عدد واشرح بإيجاز أهم أنظمة المعلومات الحاسوبية الموجهة لدعم المستويات الإدارية المختلفة؟
3. اشرح الطرق المختلفة لبناء أنظمة المعلومات الحاسوبية الموجهة لدعم عمل المجموعات؟
4. اشرح أنظمة معالجة العمليات: تعريفها ووظائفها وخصائصها الأساسية؟
5. عرف أنظمة دعم القرارات واشرح خصائصها؟
6. عدد أهم أنظمة المعلومات التي يمكن أن نجدها في منظمات الأعمال؟
7. ما هو نظام معلومات الموارد البشرية وما هي مكوناته؟
8. اشرح وظيفة ومكونات نظام المعلومات المحاسبية؟
9. اشرح وظيفة ومكونات نظام معلومات التسويق؟
10. اشرح الأنظمة الفرعية لنظام معلومات التصنيع؟
11. اشرح مفهوم نظام معلومات إدارة المواد؟
12. ارسم المخطط البيئي لنظام رقابة المخزون واشرح تدفقات البيانات فيه وحدد مصادرها ووجهاتها؟
13. ما هي وظيفة نظام المعلومات المالية وما هي الأنظمة الفرعية المكونة له؟
14. اشرح أهمية دراسة العمليات الإدارية في المنظمة وعلاقاتها بتحليل وتصميم أنظمة المعلومات؟

الفصل الثالث

أساليب ومنهجيات تطوير أنظمة المعلومات الحاسوبية

تسمى عملية بناء أنظمة المعلومات الحاسوبية بـــ " تطوير الأنظمة tem Development". وتتم عملية التطوير هذه وفقاً لمدخل حل المشكلات – ɔlem Solving Approach، أما دورة حياة تطوير النظـــام tem Development Life Cycle (SDLC) فيقصد بها سلسلة المراحل التي تمر بها عملية التطوير. وتتضمن مرحلة عادة مجموعة من الأنشطة Activities التي يتم من خلالها تحقيــــق أهـــ محددة تؤدي بالنهاية إلى بناء نظام المعلومات المطلوب.

يمكن تطوير أنظمة المعلومات باستخدام أساليب تطوير مختلفة وفقاً لطب وحجم النظام ومتطلبات استخدامه، أما منهجيات التطوير فيقصد بهــــا مجمو الأدوات المستخدمة لنمذجة النظام والطرق المتبعة لتحديد متطلبــــات المســـ (احتياجات المستخدم) وتحويلها إلى نظام معلومات حاسوبي.

وهكذا فان تطوير أنظمة المعلومات الحاسوبية يمكن أن يتم وفق أســـ متعددة، وباستخدام منهجيات تطوير مختلفة، وعلى فريق تطوير النظام أن يحدد البداية أسلوب التطوير الذي سيعتمده والمنهجية الــــتي سيســـتخدمها في ذل وسندرس في هذا الفصل الأساليب والمنهجيات المستخدمة في عملية التطوير.

1.أساليب تطوير أنظمة المعلومات

يمكن أن يتم تطوير أنظمة المعلومات الحاسوبية وفقاً لأحد الأساليب التالية:

1.1 دورة حياة تطوير النظام SDLS:

هي مجموعة من الخطوات المحددة مسبقاً والتي تمر من خلالها عملية بناء نظم المعلومات. وتستخدم هذه المنهجية لتطوير أنظمة المعلومات واضحة التحديد Well-Defined System، والتي يمكن فهمها بسهولة.

أما أهم خصائص هذه المنهجية فهي:

- تستند إلى **أسلوب حل المشكلات من أعلى إلى اسفل** Top-Down Problem Solving وهو ما يطلق عليه اسم المدخل النظمي أو الشمولي. وبناء على ذلك تبدأ عملية تطوير النظام بدراسة المشكلة (النظام الحالي) وتحديد الأهداف ثم البحث عن أفضل البدائل الممكنة لبناء النظام الجديد.
- **تطبيق إجراءات التأكد من الجودة** Quality Assurance، والتي تتضمن إجراء المراجعات اللازمة بعد كل خطوة لضمان الوصول إلى أنظمة معلومات ذات جودة عالية.
- **تدقيق مخرجات النظام** Validation للتأكد من أنها تلبي متطلبات المستخدمين User Requirements.
- **تدقيق عمليات النظام** Verification للتأكد من صحة تحويل المدخلات إلى المخرجات المطلوبة.

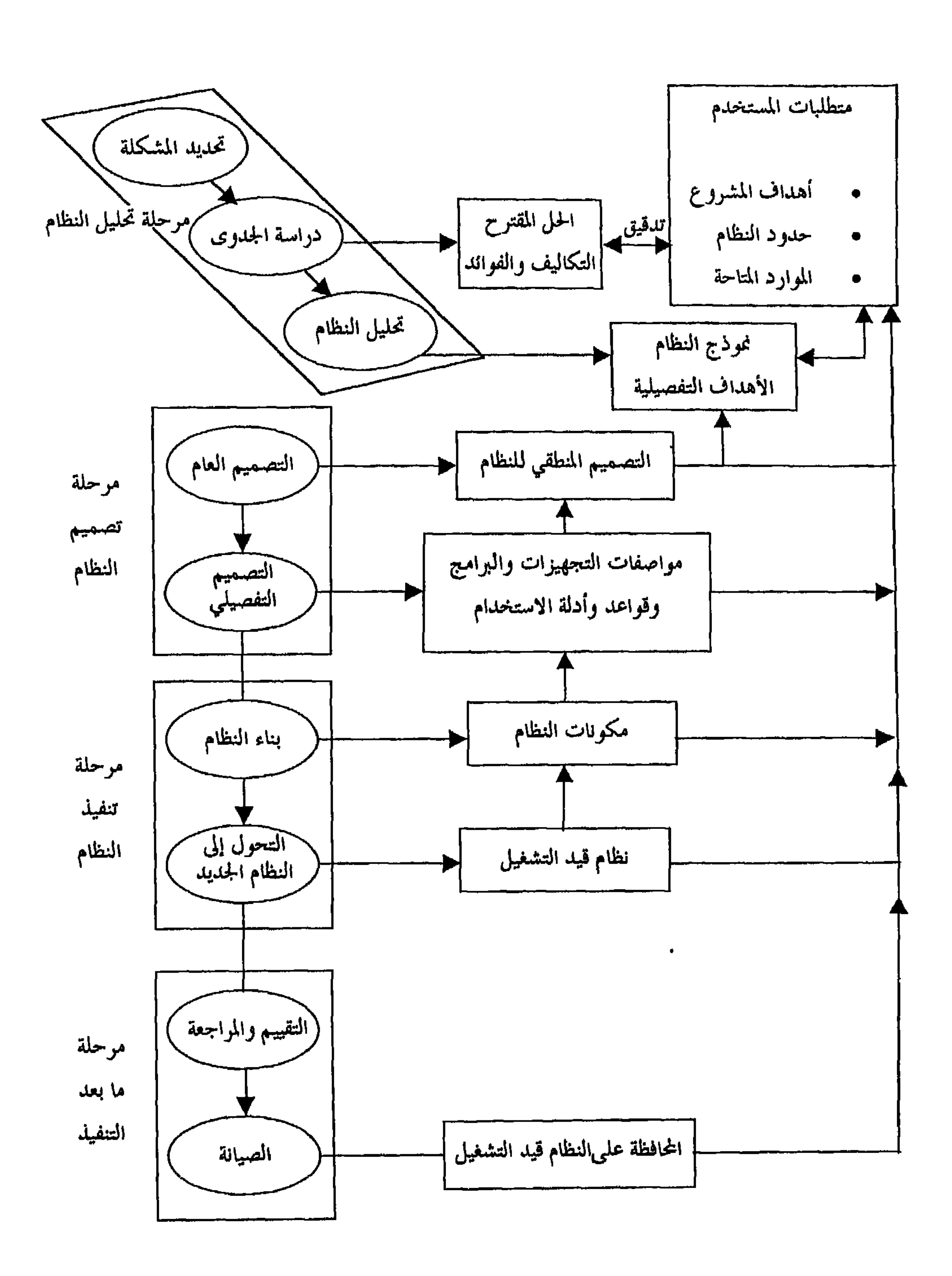
- اختبار النظام Testing للتأكد من أن مستوى أداءه يقع ضمن معايير الأداء المقبول. ويبين الجدول (1.3) وكذلك الشكل (1.3) المراحل التي تمر بها دورة حياة تطوير الأنظمة.

جدول (1.3) مراحل دورة حياة تطوير الأنظمة

المرحلة	الأنشطة/ المهام	الهدف (المخرجات)
1. تحديد المشكلة	• فهم النظام الحالي • تحديد متطلبات المستخدم	متطلبات المستخدم؛ الأهداف المطلوبة والقيود المفروضة
2.دراسة الجدوى	• اقتراح حلول بديلة للمشكلة • اختيار تحديد الجدوى المناسب من خلال الجدوى الفنية والعملية والاقتصادية لهذه الحلول	• الحل المقترح للمشكلة والتكاليف والفوائد المتوقعة منه
3. تحليل النظام (الدراسة التفصيلية للنظام)	• توضيح وتعميق فهم النظام من خلال تحديد ما يلي: - العمليات التي يقوم بها النظام. - كيفية أداء هذه العمليات. - المشاكل التي تواجه عمل المستخدم. - ما هي متطلبات المستخدم التي يجب أن تتوفر في النظام الجديد	• نموذج تفصيلي للنظام الحالي يبين الوظائف والبيانات المستخدمة فيه (النموذج الحالي المنطقي للنظام. • المتطلبات الجديدة للمستخدم.
4. تصميم النظام System Design	1. التصميم العام للنظام: - توليد مجموعة من الحلول البديلة (النماذج المادية) للنظام الجديد. - اختيار الحل الأفضل.	• النموذج المنطقي والنموذج المادي للنظام الجديد.

تابع جدول (3.1)

المرحلة	الأنشطة/ المهام	الهدف (المخرجات)
	2. التصميم التفصيلي للنظام: - تصميم قاعدة بيانات النظام. - تصميم البرامج - إعداد أدلة الاستخدام. - تصميم أسلوب التفاعل بين المستخدم و الحاسوب (واجهة الاستخدام) - تصميم إجراءات حماية النظام	• مواصفات التجهيزات والبرمجيات النظمية. • مواصفات قاعدة البيانات. • مواصفات البرامج. • أدلة الاستخدام. • إجراءات الحماية
5. تنفيذ النظام System Implementation	1. بناء مكونات النظام واختبارها. 2. التحول من النظام القديم إلى النظام الجديد: - تدريب المستخدمين - نقل البيانات - الانتقال للعمل بالنظام الجديد	• نظام معلومات قيد الاستخدام
6. ما بعد التنفيذ Post Implementation	1. المراجعة والتدقيق: تقييم مدى تلبية النظام للمتطلبات المحددة في مرحلة التحليل (إحتياجات المستخدم)	• قبول النظام أو اقتراح تعديله.
	2. الصيانة - اكتشاف الأخطاء وإزالتها - إجراء التحسينات اللازمة	• ضمان استمرارية عمل النظام



شكل (1.3) مراحل دورة حياة النظام

وبالرغم من أهمية أسلوب دورة تطوير حياة النظام، وما يوفره من خطوات متسلسلة ومحددة يتم اتباعها لضمان تطوير أنظمة معلومـــات تلـــبي الاحتياجـــات المطلوبة للمستخدمين، إلا انه يعتبر غير مناسباً لتطوير **أنظمة المعلومـــات الكبـــيرة جداً**، وكذلك تلك الأنظمة التي تتصف بعـــــدم الوضـــوح Imprecise Systems كأنظمة دعم القرارات والأنظمة الجديدة عموماً، نظراً لعدم توفر الخبرة الكافية بهـــا لحداثتها، أو لكونها لم تستخدم بعد في بيئة العمل. وهكذا فان أسلوب دورة حيـــاة تطوير الأنظمة يعتبر مثالياً وجيداً لتطوير الأنظمة التي يمكن تحديدها بدقة Precisely Defined (أي تحديد مخرجاتها وعملياتها و مدخلاتها بالدقة الكافية)، أما في الحالات التي لا يمكن الوصول فيها إلى مثل هذا التحديد الدقيق (نظراً لضخامة حجمـــها أو لعدم توفر المعلومات الكافية بمكوناتها) فانه لا بد من اللجوء إلى استخدام أســـاليب تطوير أخرى، كالتطوير التدريجي Staged approach أو استخدام النماذج التشـبيهية (التجريبية Prototypes)، أو غيرها.

2.1 التطوير التدريجي (على مراحل) Staged Development:

يستخدم هذا الأسلوب لتطوير الأنظمة الكبيرة جداً، والتي يمكن تقســيمها إلى عدد من الأنظمة الفرعية، بهدف تنفيذ عملية بناء هذه الأنظمة على مراحل، ممـــا يتيح إمكانية إدارة هذه العملية بكفاءة وفاعلية. وفقاً لهذه الطريقة يتم في كل مرحلة Stage تطوير أحد النظم الفرعية لنظام المعلومات الذي يجري تطويره، ثم يتم ربـــط هذا النظام الفرعي مع المراحل السابقة، أما خطوات تطبيق هذا الأسلوب فهي:

1) دراسة جدوى النظام الكلي (الكبير).
2) تقسيم النظام إلى عدد من الأنظمة الفرعية.

3) تقسيم المشروع إلى عدد من المراحل بحيث يتم في كل مرحلة تطوير أحد هذه الأنظمة الفرعية، باستخدام منهجية دورة حياة تطوير النظام التي تعرفنا إليها في الفقرة السابقة.

4) ربط النظام الفرعي الذي تم تنفيذه في نهاية كل مرحلة بالأنظمة الفرعية التي تم تطويرها في المراحل السابقة.

فمثلاً يمكن تقسيم نظام المعلومات في شركة كبيرة إلى عدة أنظمة فرعية مثل: نظام معلومات المبيعات ونظام معلومات المشتريات ونظام رقابة المخزون، ثم يتم في كل مرحلة بناء أحد هذه الأنظمة فقط، مما يتيح لإدارة المشروع إمكانية متابعة أنشطة التطوير والرقابة عليها بطريقة افضل. كما يفيد هذا الأسلوب في كونه يساعد على اكتساب الخبرة اللازمة بهذه العمليات لدى كل من المدراء والمستخدمين وفريق العمل الذي يقوم بالتطوير، مما يجعل المراحل التالية اكثر سهولة وكفاءة.

3.1 التطوير المعتمد على فريق عمل Team-Centered Development:

يقوم هذا الأسلوب على فلسفة مناقضة تماماً لمفهوم دورة حياة تطوير الأنظمة، حيث يترك الحرية الكاملة لفريق التطوير، لتطوير نظام المعلومات، دون أية قيود تتعلق بتسلسل عمليات التطوير. فبدلاً من التركيز على تنفيذ عمليات وأنشطة التطوير بشكل رسمي ومتسلسل كما هو الحال في دورة حياة تطوير النظام يسمح للفريق بالعمل في هذه الأنشطة بالأسلوب الذي يراه مناسباً وأكثر فعالية لإنجاز النظام المطلوب.

وتستند فلسفة هذا الأسلوب في أن عملية التطوير لا يمكن أن تتم دوماً بطريقة متسلسلة بل غالباً يتم الرجوع إلى المراحل السابقة نتيجة التوصل إلى فهم أكثر وضوحاً للنظام الذي يجري تطويره. كما أن عملية التطوير تتطلب غالباً إجراء اختبارات في المراحل المختلفة للتأكد من صحة ودقة التصورات الحالية للنظام والحلول التصميمية المقترحة لتطويره، ويمتاز هذا الأسلوب في كونه يترك لأعضاء فريق التطوير الحرية الكاملة Total free approach في العمل مما يشجعهم على التجديد والإبداع والابتكار.

4.1 التطوير الارتقائي للنظام Evolutionary Design:

يستخدم هذا الأسلوب لتطوير أنظمة المعلومات التي تكون في البداية غير واضحة التحديد Imprecise Systems، ووفقاً لذلك يتم تطوير النظام بطريقة تجريبية وبشكل ارتقائي.

ففي البداية يتم تطوير نظام أولي بسيط Pilot System يوضع قيد الاستخدام والاختبار ثم تجري ترقيته Upgrading بشكل متدرج على خطوات، حيث يتم في كل خطوة تالية إضافة إمكانات وقدرات جديدة إلى هذا النظام الموجود قيد الاختبار. وهكذا فإنه نتيجة لاستخدام هذا النظام واختباره في كل مرة، يتم اكتساب المعرفة والخبرة اللازمة لتحديد متطلبات التطوير للخطوة التالية.

يستخدم هذا الأسلوب الارتقائي في تطوير أنظمة المعلومات الحاسوبية في الحالات التالية:

أ) عندما يكون النظام الذي سيتم تطويره غير واضح Imprecise System وبالتالي يصعب تحديد متطلبات المستخدم بالدقة الكافية، ويمكن أن نصادف مثل هذه الحالات عندما يكون:

- النظام جديد كلياً وغير مستخدم حالياً.
- النظام يتضمن حوارات كثيرة بين المستخدم والحاسوب، ويصعب تحديد طريقة سير هذه الحوارات بشكل مسبق.
- النظام يقوم بحوسبة أعمال مجموعة من الأفراد Work Group يتعاونون معاً لإنجاز مهام محددة . وفي هذه الحالة يصعب أيضا الوصول إلى تحديد دقيق للتفاعلات الممكنة بين الأعضاء واللازمة لتأمين دعم فعال لعمل المجموعة.

ب) عندما يكون مطلوباً تصميم نظام لدعم القرارات Decision Support System (DSS): ففي مثل هذه الحالات تكون المشكلة التي تجري دراستها في حالة من عدم التأكد الشديد، ولا يكون واضحاً في البداية، إلى أي مدى يمكن للحاسوب أن يوفر الدعم اللازم لحل المشكلة المطلوب تصميم نظام دعم قرارات خاص بها.

5.1 التطوير بمساعدة النماذج التشبيهية (التجريبية) Prototyping

تستخدم النماذج التجريبية أو التشبيهية للمساعدة في توضيح وتحديد متطلبات المستخدم. ولقد انتشر هذا الأسلوب في التطوير نظراً للحاجة إلى تطوير أنظمة جديدة لم تستخدم من قبل، وفي مثل هذه الحالات يكون النظام المطلوب تطويره غير واضح أيضا لكل من المستخدم والمحلل على السواء. وهكذا يمكن

استخدام أسلوب النماذج التجريبية لتطوير أنظمة المعلومات الحاسوبية في الحالات التالية:

- عندما تكون الاحتياجات غير معروفة نظراً لحداثة النظام الذي يجري تطويره وعدم توفر الخبرة الكافية لتحديد هذه المتطلبات، وهذا يحدث غالباً عند استخدام تطبيقات جديدة لتقنية المعلومات.
- عندما تكون الاحتياجات معروفة ولكن المستخدم غير متأكد من إمكانية تطبيقها بشكل ميداني على أرض الواقع، ولذلك يحتاج إلى تجربتها أولا وتقييمها قبل البدء بالتطبيق الفعلي للنظام.
- عندما يكون تطوير النظام مكلفاً جداً، ولذلك يرغب المستخدم بالتأكد من صحة وجودة التصميم المقترح قبل البدء بالتنفيذ، ولذلك يمكن استخدام النموذج التجريبي لإجراء الاختبارات اللازمة والتأكد من صحة وجودة عمل النظام.
- عندما تكون المخاطرة بالانتقال إلى النظام الجديد كبيرة جداً، في هذه الحالة يمكن أولا اللجوء إلى النموذج التجريبي والتدريب على استخدامه، وإجراء الاختبارات للظروف المختلفة التي يمكن أن تحدث فيه، وبعد التأكد من توفر الإجراءات اللازمة لمواجهة الأخطار المختلفة، يمكن الانتقال إلى تطوير النظام الحقيقي (الجديد).

ويختلف أسلوب النماذج التجريبية عن أسلوب التطوير الارتقائي بأن النموذج الأولي Pilot System الذي يتم إعداده خلال عملية التطوير يتحول في النهاية ليصبح جزءاً من النظام النهائي الذي يجري بناءه. أما في أسلوب النماذج

التجريبية فان هذا النموذج يتم استبداله في النهاية بالنظام الأصلي، أي يستخدم فقط كأداة للتحليل وتحديد الاحتياجات.

أما أهم مجالات استخدام النماذج التجريبية فهي:

- إختبار الحلول التصميمية الجديدة بشكل عملي.
- تحليل المتطلبات Requirement Analysis باعتبارها أداة هامة تساعد في ذلك توضيح متطلبات المستخدم.
- عرض الإمكانات والخيارات التي يمكن أن يوفرها النظام المقترح.

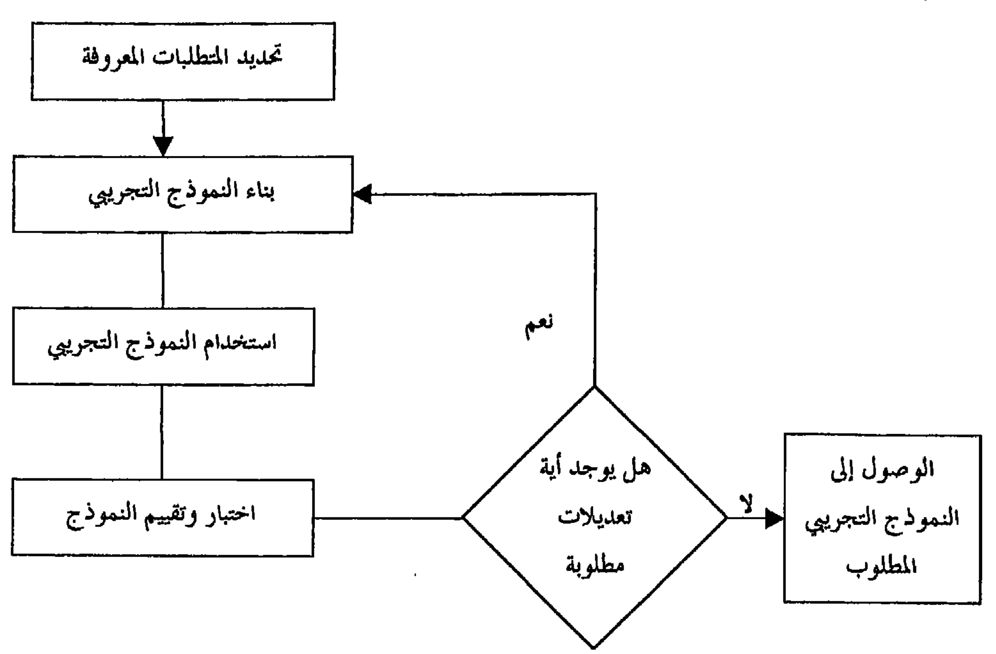
كما يمكن استخدام النماذج التجريبية ضمن دورة حياة النظام في مرحلة دراسة الجدوى للأغراض التالية:

- لإجراء الاختبارات بهدف الوصول إلى فهم أفضل عن الحلول الممكنة للمشكلة التي تجري دراستها.
- لتحديد العمليات الرئيسة للنظام المطلوب تطويره.
- لتقييم البدائل المختلفة المقترحة لتصميم النظام.

أما في مرحلة التصميم فتستخدم النماذج التجريبية لتعريف المستخدم بشكل ملموس على مدخلات ومخرجات النظام المقترح وعلى طريقة عمله. ويبين الشكل (2.3) خطوات تطبيق هذا الأسلوب والتي تشمل:

- تحديد المتطلبات المعروفة
- تطوير النموذج التجريبي واستخدامه لفترة من الزمن للتعرف على إمكاناته ووظائفه واكتساب الخبرة في التعامل مع مدخلاته ومخرجاته.
- تقييم النموذج التجريبي بهدف اقتراح الخصائص التي يجب حذفها منه، أو تلك التي يجب تعديلها، أو إضافة خصائص جديدة وهكذا.

- إجراء التعديلات المطلوبة في النموذج التجريبي ثم وضعه قيد الاستخدام وتقييمه، وهكذا حتى الوصول إلى الصيغة النهائية له كما هو مبين في الشكل (2.3) أدناه.

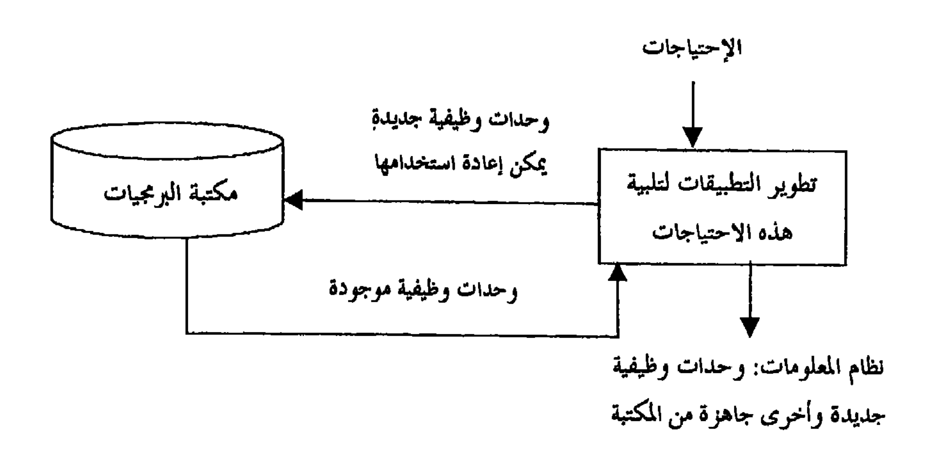


الشكل (2.3) خطوات استخدام النماذج التجريبية

6.1 أسلوب التركيب Synthesis:

وفقاً لهذا الأسلوب يتم بناء (تطوير) أنظمة المعلومات من وحدات وظيفية (Modules) موجودة، وهذا يتطلب أن يتوفر لدى المنظمة مكتبة برمجيات Software Library تضم العديد من الوحدات الوظيفية Modules الجاهزة والتي يمكن إعادة استخدامها مرات عديدة Reusable عند بناء أنظمة المعلومات الجديدة، ولكي تتوفر مثل هذه المكتبة يجب أن تعتمد المنظمة الأسلوب الهيكلي لتصميم النظم Structural Design، الذي يقوم على تقسيم النظام إلى مجموعة من الوحدات الوظيفية

Program Modules التي تقوم كل منها بوظيفة محددة تماماً. وأهم مميزات هذا الأسلوب هو **إعادة استخدام** هذه الوحدات الوظيفية عند تصميم الأنظمة الجديدة، والشكل (3.3) يوضح هذا الأسلوب لتطوير الأنظمة.



شكل (3.3) نظام المعلومات يمكن أن يحتوى العديد من الوحدات الوظيفية الجاهزة.

7.1 اختيار الأسلوب المناسب لتطوير النظام:

يعتمد اختيار الأسلوب المناسب لتطوير نظام المعلومات الحاسوبية على عدد من العوامل أهمها:

أ) **درجة وضوح وتحديد بنية النظام:** فالنظم ذات البنية الواضحة Well-Defined يمكن تطويرها باستخدام أسلوب دورة حياة تطوير النظام المشار إليها أعلاه، أما بالنسبة للأنظمة التي تكون بنيتها غير واضحة فيمكن تطويرها باستخدام الأسلوب التدريجي أو النماذج التجريبية.

ب) **توفر الخبرة والمعرفة بتقنية الأنظمة:** أي عندما يكون المستخدمين على معرفة واطلاع بتقنية المعلومات، يمكن تطوير النظام باستخدام أسلوب دورة حياة النظام، أما عندما يكون المستخدمين غير عارفين بهذه التقنية ويستخدمونها للمرة الأولى فانه يفضل تطوير النظام باستخدام الأسلوب التجريبي.

جـ) **حجم النظام الذي يجري تطويره** عندما يكون النظام كبير جداً، يفضل تجزئته إلى نظم فرعية محددة بشكل واضح ثم اتباع الأسلوب المرحلي Staged لتطويره، أي التطوير على مراحل.

2. منهجيات تطوير الأنظمة System Development Methodologies

يقصد بمنهجية التطوير مجموعة الطرق والأدوات المستخدمة في تطوير النظام بدءاً من تجميع المعلومات وتحديد احتياجات المستخدم، مروراً بعمليات النمذجة ووصولاً إلى بناء النظام الحاسوبي الجديد بالطريقة التي تلبي الاحتياجات بالكفاءة والفاعلية المطلوبة.

وهكذا فان المنهجية تحدد الخطوات التي سيتم من خلالها تطوير النظام والأدوات Tools اللازمة لكل خطوة، ونظراً للتنوع الشديد في أساليب تطوير النظم ووجود العديد من الأدوات التي يمكن استخدمها في كل خطوة أو مرحلة من عملية التطوير، فانه نظرياً يمكن اشتقاق العديد من المنهجيات الممكنة لتطوير النظم، ولكن من الناحية العملية يفضل أن تستخدم المنظمات منهجية واحدة عامة لتطوير أنظمتها Standard Methodology نظراً للمزايا التالية:

- تنميط وتوحيد عمليات وأنشطة التطوير لكي تتم بأسلوب موحد ومنسق مما يسهل عمليات التوثيق والمتابعة.
- تركيز جهود المحللين والمصممين على دراسة النظم التي يقومون بتطويرهـــا، وإعفاءهم من المهام المتعلقة باختيار الأدوات والأساليب المختلفـــة للنمذجـــة والتوثيق لكل مشروع على حدة.
- اعتماد منهجية موحدة يساعد المحللين والمصممين ومساعديهم وبقية أعضـــاء فريق التطوير في إتقان استخدام أدوات وأساليب هذه المنهجية، كمـــا يجعـــل المستخدمين اكثر فهماً لها، مما يوفر الكثير من الوقت والجهد اللازم للتفاعل مـــع هؤلاء الأفراد في حالة استخدام منهجيات تطوير مختلفة لكل مشروع.

ويقصد بتحديد منهجية معينة، واعتمادها لتطوير أنظمة المعلومات في المنظمة مـــا يلي:

- **تحديد مجموعة الأدوات والأساليب** التي سيتم استخدامها في إنجاز المـــهام المختلفة بدءاً من تجميع المعلومات وإنتهاءا بتنفيذ النظام ووضعه قيد التشغيل.
- توزيع استخدام هذه الأدوات والأساليب على **سلسلة من الخطوات الـــتي سيتم وفقها تطوير النظام**، وبحيث تكون مخرجات كل خطوة هي مدخلات للخطوة التالية. وهذا يعني تحديد الخطوات التي سيتم مـــن خلالهـــا تنفيـــذ مراحل دورة حياة التطوير، والأدوات التي ستســـتخدم في كـــل مرحلـــة، والطريقة التي سيتم استخدامها لتوثيق النظام.

وبشكل عام، فان جميع منهجيات التطوير تؤدي في النهاية إلى بنـــاء نظـــام المعلومات المطلوب، ولكن اختلاف هذه المنهجيات يعود إلى اختلاف الطريقة الـــتي تعتمدها كل منهجية لتنفيذ مهام وأنشطة عملية التطوير، ونظـــراً لان المكونـــات

الرئيسة لأي نظام هي عملياته وهيكل بياناته وتدفقات البيانات فيه، فان المنـهجيات المختلفة يمكن أن تبدأ بدراسة إحدى هذه المكونات وتنطلق منها لبناء نموذج النظـام وتطويره. ولذلك نجد أن بعض المنهجيات تركز على دراسة تدفقات البيانات، بينما تركز منهجيات أخرى على دراسة العمليات، في حين تركز منهجيات ثالثة علـــى دراسة هيكل بيانات النظام.

في ضوء ما سبق، يمكن تصنيف منهجيات التطوير إلى:

- **منهجيات تركز على دراسة تدفقات البيانات في النظـــام Data-Flow Based Methodologies**، حيث تبدأ بدراسة هذه التدفقات ثم تقوم بتحديـــد مصادرها ووجهاتها هذه التدفقات (العمليات وأماكن تخزين البيانات).
- **منهجيات تركز على دراسة بيانات النظـام** Data-Based Methodologies حيث تبدأ بدراسة نموذج بيانات النظـام Data Model ثم تنتقـــل إلى دراســة المكونات الأخرى للنظام.
- **منهجيات تركز على دراســة وظــائف النظــام** Functional Analysis Methodology، وهي تركز على دراسة وظائف النظام وعملياته ثم تنتقـــل إلى تحديد تدفقات البيانات اللازمة لهذه العمليات والناتجة عنها وتحديد مصادرهـــا ووجهاتها.

ومن الواضح أن أي من هذه المنهجيات قد لا تعطي لوحدها تصوراً كاملاً عن النظام، ولذلك يزداد استخدام أدوات من اكثر من منهجية واحـــدة لنمذجــة النظام. مما سيؤدي إلى التوجه نحو دمج هذه المنهجيات في منهجية واحدة بفضـــل التطورات المستمرة في أدوات هندسة البرمجيات بمساعدة الحاســـوب (-Computer

التي ساعدت في أتمتة العديد من عمليـــات Aided Software Engineering (CASE التحليل والتصميم، والتي سنتعرف عليها في فصل لاحق في هذا الكتاب.

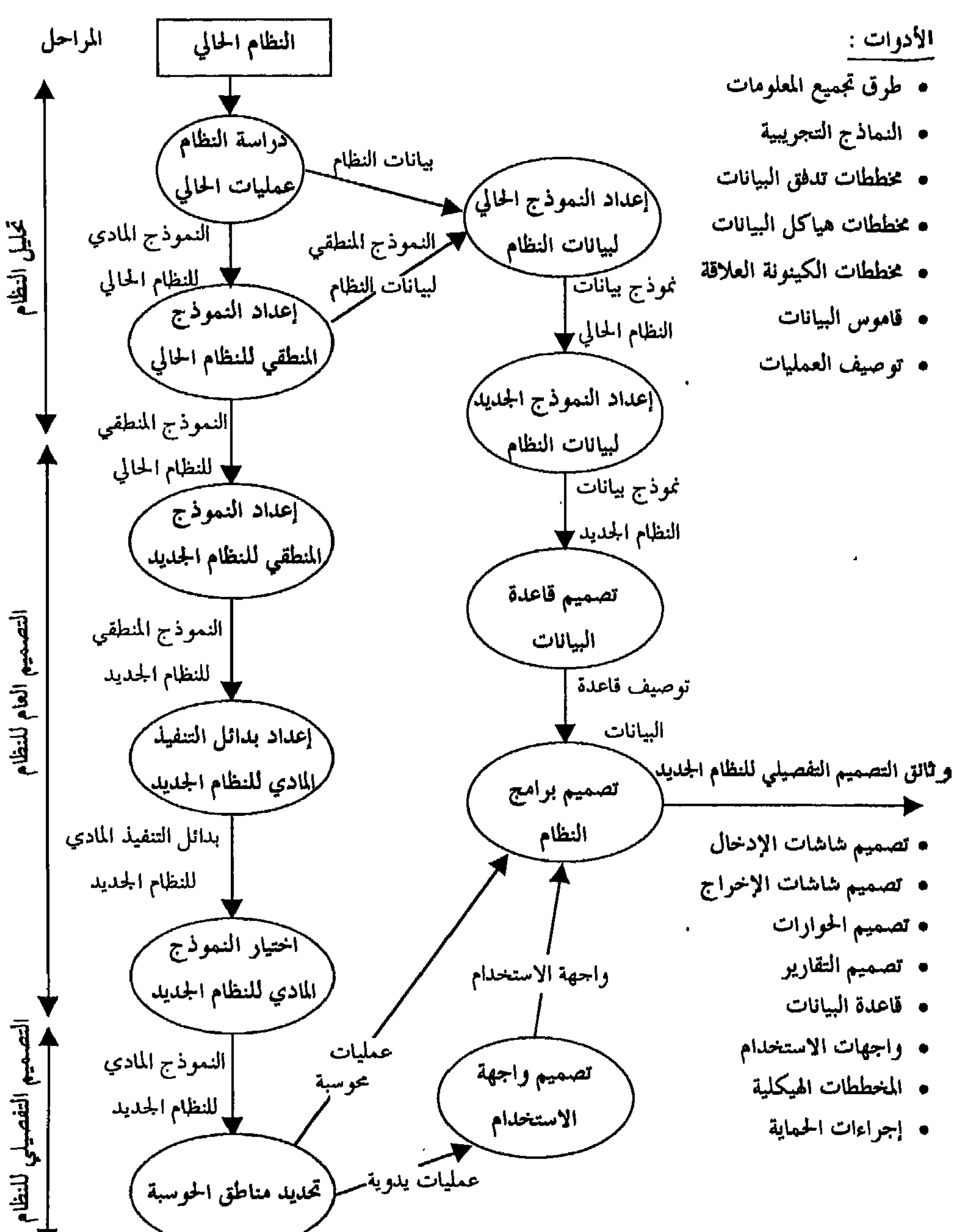
أما الآن فسنتعرف إلى أهم المنهجيات المستخدمة في تطوير الأنظمة والـــتي سنعتمد عليها بشكل كبير في دراستنا لهذا المقرر.

1.2 منهجية التحليل الهيكلي Structured System Analysis

تم تطوير هذه المنهجية في نهاية السبعينات وتعتمد على اســتخدام الأدوات الهيكلية كمخططات تدفق البيانـات Data Flow Diagrams (DFD) والتوصيــف الهيكلي للعمليات Structured Process Specification .

أما نمذجة البيانات فتتم من خلال وصفها بشكل علاقات (جــــداول) ثم إجراء عمليات التسوية Normalization اللازمـــة للوصـــول إلى النمـــوذج الحالي لبيانـــات النظام. وهكذا يتم إجراء الربط اللازم بين التحليـــل الوظيفـــي للنظـــام وتحليـــل البيانات فيه على النحو المبين في الشكل (4.3).

نلاحظ من الشكل (4.3) أعلاه أن هذه المنهجية لا تقتصر علـــى تحديـــد الخطوات التي سيتم من خلالها الانتقال من النظام الحالي إلى النظـــام الجديـــد، بـــل تتضمن أيضا الأساليب والأدوات التي يجب استخدامها في كل خطـــوة، وكيفيـــة تكامل استخدام هذه الأدوات في تحليل مشكلات النظام وإيجاد الحلول المناسبة لهـــا. فالنموذج المادي والنموذج المنطقي للنظام الحالي يتم وضعهما خلال مرحلة تحليـــل النظام، ومخرجات هذه المرحلة تكون عادة مخططات تدفـــق البيانـــات في النظـــام الحالي،وفي هذه المرحلة أيضا يتم إعداد مخطط البيانات في النظام الحالي. وتكون هـذه المخططات بمثابة مدخلات إلى مرحلة التصميم العام التي يجري فيها إعداد النموذج



شكل (4.3) خطوات منهجية التحليل الهيكلي

المنطقي ثم المادي للنظام الجديد في ضوء أهداف المشـــروع والمتطلبـــات الـــتي تم تحديدها في مرحلة التحليل السابقة. كما يتم في هذه المرحلة إدخـــال التعديـــلات اللازمة في نموذج بيانات النظام بما يتماشى مع متطلبات النموذج المنطقـــي للنظـــام الجديد. أما مخرجات مرحلة التصميم العام فهي: النموذج المادي للنظـــام الجديـــد" ونموذج البيانات الجديد. وهذه المخرجات تمثل مدخلات للمرحلة التاليـــة وهـــي مرحلة التصميم التفصيلي للنظام، والتي يتم خلالها تقسيم النموذج المـــادي للنظـــا الجديد إلى عمليات ستخضع للحوسبة، وعمليات يستمر أداءها بطريقة يدوية. ويتـــ إعداد التصاميم الخاصة بواجهة الاستخدام، وكذلك تصاميم برامج النظام وتوصيـــف قاعدة البيانات وغيرها من أنشطة التصميم التفصيلي الأخرى.

2.2 منهجية التحليل والتصميم الهيكلي للنظم Structured System Analysis and Design Method (SSADM)

تم تطوير هذه المنهجية وأصبحت بمثابة منهجية قياسية في قطاع الخدمـــا المدنية في بريطانيا، وتعتمد فلسفة هذه المنهجية على نفس المبادئ التي تعرفنا إليـــها الفقرة السابقة، إلا أنها تجمع كل من عملية تحليل البيانات وتحليل التدفقـــات في آ واحد. وتمتاز هذه المنهجية بكونها تحدد بدقة وبالتفصيل الكامل جميع المهام المطلوبـ والتسلسل اللازم لتنفيذها ضمن كل مرحلة من مراحل تطوير النظام.

تقسم هذه المنهجية دورة تطوير النظام إلى ثلاث مراحل رئيســـة تتـــوز بدورها إلى ثمانية مراحل فرعية على النحو المبين في الجدول (2.3). وتتكـــون كـــ مرحلة فعلية من مجموعة محددة من المهام Tasks. ويتم وصف كل مهمة من خـــلا مدخلاتها ومخرجاتها وأدوات تنفيذها.

المراحل الرئيسة	المراحل الفرعية	الأدوات والأساليب
1. توصيف المشكلة	1. تحديد المشكلة 2. تعريف المشروع	• طرق تجميع المعلومات • المخططات العامة لتدفق البيانات وهياكل البيانات
2. التحليل	1. تحليل النظام الحالي 2. توصيف المتطلبات 3. التقاء البديل التقني	• مخططات E-RD,DFD • توصيف العمليات • قاموس البيانات • دورة حياة الكينونات • التصميم العام للحوارات • معايير قبول النظام
3. التصميم	1. تصميم البيانات 2. تصميم العمليات 3. التصميم المادي	• تحليل العلاقات • توصيف قاعدة البيانات • المخططات الهيكلية • تصاميم الحوارات • تصاميم المدخلات والمخرجات • تصميم نماذج النظام

جدول رقم (2.3) المراحل الرئيسة والفرعية لمنهجية SSADM

نلاحظ من الجدول أعلاه أن هذه المنهجية تعتمد أسلوب دورة حياة تطوير النظام التي تعرفنا إليها في بداية هذا الفصل حيث تبدأ بتحديد المشكلة ثم دراسة الجدوى وتعريف المشروع، يلي ذلك مراحل التحليل فالتصميم.

3.2 منهجية هندسة المعلومات Information Engineering Methodology (IE)

تم وضع أسس هذه المنهجية الجديدة لتطوير أنظمة المعلومات في بداية الثمانينات للتغلب على المشاكل الناتجة عن استخدام المنهجيات الأخرى والتي نسميها بالمنهجيات التقليدية المعتمدة على الإجراءات Procedure-Driven، كالتكاليف المرتفعة لصيانة النظام، وضرورة إدخال التعديلات أو التحسينات فيه، والاحتمالية الكبيرة لحدوث الأخطاء المختلفة خلال عمليات التحليل أو التصميم، والتي يمكن أن تنتج عن سوء الفهم أو عن عدم وجود مفاهيم موحدة بين المستخدمين والأخصائيين الذين يقومون بتطوير النظام. والمشكلة الأهم من ذلك كله هي إمكانية عدم توفر البيانات المطلوبة لاتخاذ قرار معين نظراً لكون هذا القرار يتخذ للمرة الأولى في المنظمة، أي أن قاعدة البيانات لا تتضمن المدخلات اللازمة لتوفير المعلومات المطلوبة لذلك القرار.

ولذلك تنطلق هذه المنهجية، التي يمكن اعتبارها Data-Driven من التخطيط الاستراتيجي للمنظمة ككل، وبناء نموذج البيانات الذي يأخذ بعين الاعتبار ليس فقط الاحتياجات الحالية بل المستقبلية للبيانات. وهكذا يتم بناء هذا النموذج في ضوء الخطط الاستراتيجية للمنظمة وبالتعاون الوثيق مع الإدارة العليا لها، فهي وحدها القادرة على التنبؤ بالاحتياجات المستقبلية.

وتتطلب هذه الاستراتيجية من المحللين دراسة مجال العمل Business Area الذي تمارسه المنظمة واستكشاف التحديات التي تواجهها، ومعرفة بيئتها واتجاهات التطوير المحتملة فيها، كما تعتمد هذه المنهجية ضرورة أتمتة جميع عمليات التطوير لتسريع دورة حياة النظام وتحسين كفاءتها وفاعليتها.

تتم هندسة المعلومات من خلال برمجيات خاصة تكون بمثابة نظم خبيرة محوسبة، تساعد كل من المدراء وكذلك المحللين والمصممين في تحليل إحتياجات العمل، وتنفيذ عمليات التحليل والتصميم وتوليد قواعد البيانات بصورة آلية، وهذا كله يؤدي في النهاية إلى توليد نظام المعلومات المطلوب بسرعة وكذلك توليد نظام خبير في قطاع الأعمال الذي تمارسه المنظمة.

وهكذا فان منهجية هندسة المعلومات تضيف إلى دورة حياة تطوير النظام مرحلة هامة جديدة هي التخطيط الاستراتيجي، التي يتم خلالها تحديد مشاريع أتمتة عمليات تطوير الأنظمة انطلاقاً من الخطط الاستراتيجية لعمل المنظمة، وهكذا تستخدم منهجية هندسة المعلومات لتطوير التطبيقات الحاسوبية المختلفة، وأنظمة دعم القرارات والأنظمة الخبيرة في مجال قطاع الأعمال الذي تمارسه المنظمة.

وتتضمن منهجية هندسة المعلومات، أربعة مراحل أساسية هي:

1. **تخطيط أنظمة المعلومات Information System Planning:**

في هذه المرحلة يتم دراسة خطط المنظمة وتوجهاتها و اعداد النموذج العام لها Enterprise-Wide Model ثم يتم تقسيم هذا النموذج إلى مجالات أو وحدات الأعمال المكونة له.

2. **تحليل مجالات الأعمال Business Area Analysis:**

يتم خلال هذه المرحلة إعداد النموذج المنطقي لوحدات أو قطاعات الأعمال المختلفة، وذلك باستخدام مخططات تدفق البيانات ومخططات الكينونة-العلاقة وغيرها من أدوات النمذجة. مما يسمح للمصمم تحديد المشاكل والقضايا الأساسية التي يمكن أن يواجهها كل قطاع من قطاعات أعمال المنظمة، ثم يتم

تصميم نموذج المعلومات الخاص بكل قطاع وتحديد المتطلبات الفنية للأنظمة الجديدة ووضع خطط تنفيذها.

3. تصميم الأنظمة System Design:

يتم خلال هذه المرحلة تحديد المتطلبات التصميمية ووضع التصاميم التفصيلية للأنظمة الجديدة، بما في ذلك واجهات استخدامها ومتطلبات ربطها مع الأنظمة الأخرى في المنظمة وخطط اختبارها.

4. التشييد Construction:

وهي المرحلة الأخيرة في منهجية هندسة المعلومات ويتم خلالها توليد واختبار برامج النظام وإنشاء قاعدة البيانات وتدريب المستخدمين ووضع النظام الجديد قيد التشغيل.

وأخيراً تجدر الإشارة إلى توفر العديد من الأدوات التي تدعم استخدام منهجية هندسة المعلومات في تطوير أنظمة المعلومات الحاسوبية، كأدوات هندسة البرمجيات بمساعدة الحواسيب CASE Tools و محطات عمل هندسة المعلومات Information Engineering Workbench وتسهيلات هندسة المعلومات IE Facility والتي سنتعرف عليها بشكل تفصيلي في فصل لاحق من هذا الكتاب.

4.2 منهجية الاستخدام المرن لأدوات وأساليب التطوير:

نلاحظ في المنهجيات السابقة أنها تقيد المصمم والمحلل باستخدام أدوات معينة في كل مرحلة من مراحل تطوير النظام، ولذلك فان المنهجية المقابلة لهذه المنهجيات تقوم على ترك الحرية الكاملة للقائمين على تطوير النظام لاستخدام التقنيات والأدوات والأساليب التي يروها مناسبة لتطوير النظم التي يعملون على بناءها. وتسمى هذه المنهجية بالمرنة أو منهجية الاستخدام المرن لأدوات التطوير

Flexible Use of Tools، ويمكن استخدام هذه المنهجية في أسلوب التطويـــــر مـــن خلال النماذج التجريبية، أو في التطوير الارتقائي للنظم. وبـــالرغم مـــن أن هـــذه الاستراتيجية يمكن أن تبدو مغرية ومشجعه إلا أنها تتطلب استخدام أدوات وتقنيات متكاملة مع بعضها البعض، وهذا قد لا يتوفر في حــــالات عديــدة، فالبيانـــات أو المعلومات التي يتم تجميعها أو نمذجتها باستخدام أداة معينة، يجب أن تنقل في المراحل التالية إلى أدوات أخرى، ولذلك لا بد أن تتوفر إمكانية التكامل بين هــذه الأدوات، أي سهولة تحويل النماذج التي تصف النظام من أداة إلى أخرى: مثلا تحويل مخططات تدفق البيانات إلى مخططات هياكل بيانات أو تحويل مخطط البيانات E-RD إلى نموذج علاقاتي وهكذا. وتجدر الإشارة إلى أن أدوات هندسة البرمجيات بمساعدة الحاســوب CASE توفر إمكانيات جيدة في هذا المجال مما يدعم استخدام هذه المنهجية المرنة.

5.2 منهجية استخدام الحزم البرمجية الجاهزة:

نظراً لكون العديد من التطبيقات الإدارية ذات طبيعة متماثلة في مختلــــف المنظمات، فلقد انتشرت وبشكل واسع حزم برمجيات تطبيقية جاهزة توفر إمكانيات جيدة لحوسبة مجموعة واسعة من وظائف المحاسبة والتخطيط المــالي والتنبــؤ وإدارة الإنتاج والمخزون وتحليل المبيعات ومعالجة الطلبات وإصدار أوامر الشراء والتطبيقات المتعلقة بقطاعات أعمال معينة كالبنوك وشركات التأمين وشركات النقــــل وإدارة المشافي والمؤسسات الصحية والشركات الصناعية والمؤسسات التعليميــة وغيرهـــا. ولقد وفرت هذه الحزم الجاهزة إمكانية حوسبة التطبيقات المختلفة بسرعة وبكفــاءة ملحوظة. وعند استخدام هذه المنهجية لتطوير الأنظمة الحاسوبية فانه من المهم جــداً أن يكون المستخدم أو الجهة المستفيدة على دراية واسعة باحتياجاتها تمكنـــها مـــن

اختيار الحزمة المناسبة ووضعها قيد التشغيل، والاستفادة منها بأسرع وقت وبدون الحاجة إلى إجراء تغييرات أو تعديلات واسعة فيها.

وبشكل عام يتزايد الإقبال على استخدام الحزم الجاهزة في حوسبة التطبيقات والنظم المختلفة، نظراً للتطور السريع والكبير في الإمكانات التي توفرها هذه الأنظمة في معالجة البيانات، ولما تتيحه من مرونة كبيرة في إدخال التعديلات والتغييرات، وتفضيلات المستخدم بما يتماشى مع التغيرات الممكنة في الإحتياجات، ولكن يبقى استخدام هذه الحزم محصوراً أو محدداً بالتطبيقات النمطية ذات الطبيعة العامة. أما التطبيقات الخاصة بكل منظمة فلا بد من تطويرها وفق المنهجيات التي تعرفنا إليها أعلاه.

أسئلة الفصل

1. ما هو الفرق بين أساليب التطوير ومنهجيات التطوير؟
2. عدد خصائص ومزايا استخدام أسلوب دورة حياة تطوير النظام لتطوير أنظمة المعلومات.
3. عدد مراحل دورة حياة تطوير النظام واشرح الأنشطة التي تتضمنها كل مرحلة.
4. اشرح أسلوب التطوير التدريجي (على مراحل) لنظام المعلومات والحالات التي يستخدم فيها.
5. اشرح أسلوب التطوير المعتمد على الفريق والفلسفة التي يقوم عليها، والحالات التي يمكن أن يستخدم فيها.
6. قارن بين أسلوب التطوير الارتقائي والنماذج التجريبية المستخدمان لتطوير أنظمة المعلومات.
7. اشرح الحالات التي يمكن فيها استخدام النماذج التجريبية.
8. اشرح خطوات تطبيق أسلوب النماذج التجريبية.
9. اشرح أسلوب التركيب من الوحدات الجاهزة وما هي الفلسفة التي يقوم عليها.
10. اشرح العوامل المختلفة التي يجب دراستها عند اختيار الأسلوب المناسب لتطوير نظام المعلومات.

11. ما المقصود بمنهجيات تطوير أنظمة المعلومات؟ وما هي مزايـــا اســتخدام منهجية موحدة في المنظمة؟

12. كيف يمكن تصنيف المنهجيات المختلفة لتطوير أنظمة المعلومات؟

13. اشرح خطوات منهجية التحليل الهيكلي للنظام.

14. اشرح منهجية هندسة المعلومات والمراحل الرئيسية الأربعة المكونة لها؟

15. ما هو المقصود بمنهجية الاستخدام المرن لأدوات وأساليب تطوير الأنظمة؟

16. اشرح منهجية استخدام الحزم البرمجية الجاهزة ومزاياها؟

تحليل أنظمة المعلومات الحاسوبية



الفصل الرابع

تجميع المعلومات

1- مقدمة

إن الفهم الجيد للنظام وتحديد مشاكله ومتطلبات تحديثه يعتبر عاملا هاما وشرطا رئيسيا لضمان جودة النظام الذي يجري تطويره . ومن هنا تأتي أهمية البحث عن استراتيجية فعالة تضمن تجميع الحقائق والمعلومات ذات العلاقة بعمل النظام . وبشكل عام فإن الفهم الجيد للنظام يساعد محلل الأنظمة في :

- التحديد الدقيق لمشاكل النظام الحالي .
- اقتراح حلول عملية فعالة للتغلب على هذه المشاكل.
- تلبية احتياجات جميع الأشخاص والجهات ذات العلاقة بعمل النظام.

وللوصول إلى الفهم الجيد لعمل النظام يجب أن يبذل المحلل جهداً متواصلاً لتجميع المعلومات المتعلقة بالنظام من خلال :

- إجراء العديد من المقابلات الشخصية للأفراد الذين سيتعاملون معه أو يستخدمونه .
- القيام بالملاحظات الشخصية الميدانية ودراسة سير الإجراءات الحالية والوثائق المختلفة المستخدمة فيها .
- الحصول على نسخ من جميع وثائق المدخلات والمخرجات كالتقارير والكشوفات والجداول والنماذج المستخدمة في النظام الحالي .

- دراسة وثائق النظام المتعلقة بالإجراءات الحالية والبرامج الحاسوبية إن وجدت والملفات التي تستخدمها وواجهات التعامل معها .
- دراسة البرمجيات الموجودة وكذلك التجهيزات الحالية وفحص أدائها وطاقاتها ومراجعة السجلات المتعلقة بها ، ودراسة المجلات والنشرات والكاتالوجات المتخصصة .

ونظراً لأهمية العملية وتأثيرها المباشر في جودة عمليات التحليل والتصميم اللاحقة، فإننا سنتعرف في هذا الفصل على **المصادر** Information Sources التي يمكن أن يستخدمها محلل الأنظمة للحصول على المعلومات **والطرق** Search Methods المناسبة للتعامل مع هذه المصادر، وكذلك تحديد **خطة البحث** What To Study اللازمة لتجميع المعلومات **واختيار الأساليب المناسبة لنمذجة النظام** Modeling Techniques الذي يجري تحليله .

2- الإطار العام لتجميع المعلومات

نظراً لأهمية هذه العملية ولما تتطلبه من جهد ووقت كبيرين ، فإنه يجب أن تتم بطريقة منظمة ووفق خطة مدروسة . ويطلق على هذه الخطة استراتيجية البحث Search Strategy .

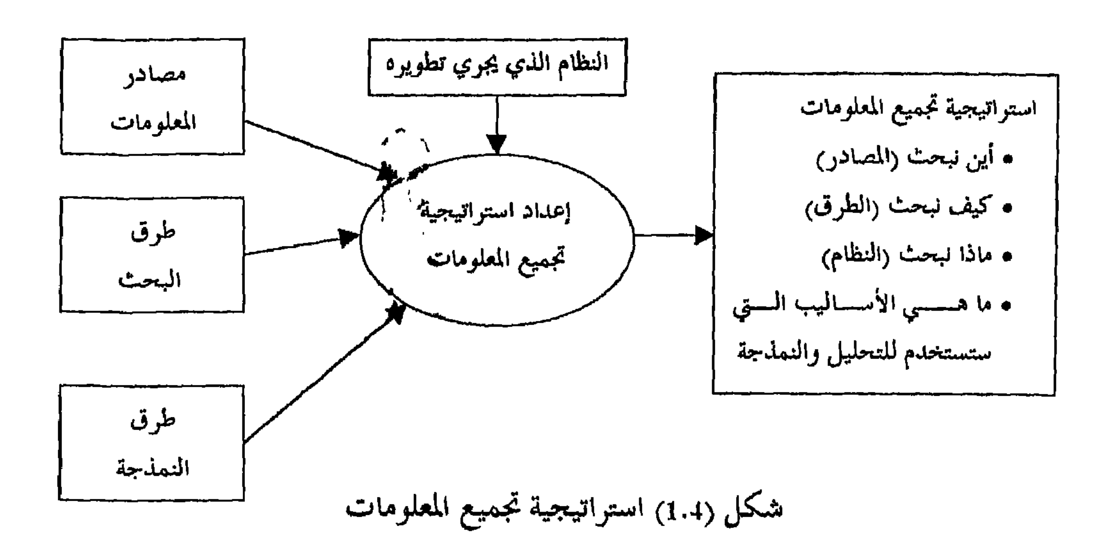
وتعتبر هذه الاستراتيجية في غاية الأهمية حيث تحدد الخطوات التي يجب اتباعها لتجميع المعلومات اللازمة لتحليل وتصميم النظام . وتتضمن عادة خطة البحث التي يجب أن تتم من الأعلى للأسفل ، أي تبدأ بتجميع المعلومات المتعلقة بالنظام ككل (أهدافه العامة ووظائفه ومدخلاته ومخرجاته وعناصر البيئة الخاصة به) ثم تتابع هبوطاً نحو تحديد مكونات النظام والتعرف على وظائفها وعلاقاتها مع

بعضها البعض في إطار النظام. يلي ذلك تحديد العناصر التي تشكل هذه المكونـــات وما تتضمنه من عمليات وما تتطلبه هذه العمليات من مدخلات وما ينتج عنها من مخرجات.

وبشكل عام تتضمن استراتيجية البحث تحديد العناصر التالية :

- **مصادر المعلومات Information Sources**: التي سيتم الحصول منها على المعلومات المتعلقة بالنظام الذي تجري دراسته .
- **طرق البحث Search Methods**: التي سيتم اســـتخدامها لتجميـــع المعلومات من هذه المصادر .
- **خطة البحث Search Plan**: وتتضمن المواقع أو العناصر أو الأحداث والعمليات التي يجب أن يتم تجميع المعلومات الخاصة بها . أي النظام الذي ستجري دراسته بهدف فهم عملياته وتحديد احتياجاته .وهذا يشمل تحديد **العمليات الحالية** في النظام، والمهام Tasks التي تتضمنها هذه العمليـلت، والأدوات المستخدمة في إنجازها **والعلاقات المتبادلة** فيما بينها.
- **الأساليب Modeling Techniques**: التي سيتم استخدامها للاحتفاظ بالمعلومات التي سيتم تجميعها ، وكذلك لتحليل هذه المعلومات . وهـــذه يمكن أن تتراوح بين توثيق المعلومات بشكل ملفات يدوية وبـــين بنـــاء النماذج Modeling التي تعكس عمل النظـــام ، كمخططـــات تدفـــق البيانـــات ومخططات هياكل البيانات وغيرها من أدوات النمذجة .

ويوضح الشكل (1.4) مكونات استراتيجية تجميع المعلومات .



شكل (1.4) استراتيجية تجميع المعلومات

3- مصادر المعلومات Information Sources

يستخدم محلل النظم عادة مصادر متنوعة للحصول على المعلومات المتعلقة بالنظام الذي تجري دراسته . وفيما يلي عرض لأهم هذه المصادر :

1.3 مستخدمي النظام System Users :

يعتبر المستخدمين الحاليين أو المحتملين من أهم مصادر المعلومات المتعلقة بالنظام . فمن خلالهم يستطيع المحلل الوصول إلى فهم موضوعي للنظام ومشاكله . ويستخدم المحلل أساليب المقابلات الشخصية والاستبيانات للحصول على المعلومات المتعلقة بالأهداف ، والمتطلبات اللازمة لتحقيق هذه الأهداف ، وكذلك التعرف على الأنشطة اللازمة لذلك .

2.3 الوثائق Documents

تعتبر الوثائق المستخدمة في النظام مصدراً هاماً للحصول على المعلومات المتعلقة بحركة البيانات Data flows والعمليات Transactions التي يقوم بها النظام . ويبدأ محلل الأنظمة عادة بإعداد قائمة كاملة لجميع النماذج والوثائق المستخدمة في

النظام (الفواتير – أوامر الشراء – مذكرات الاستلام – الميزانية – سندات القيد وغيرها). ثم يقوم بدراسة محتوى هذه الوثائق واحدة بعد أخرى. ويطلق على هذه العملية اسم تحليل المحتوى Content Analysis، ويتم من خلالها دراسة الفقرات المختلفة للنموذج أو الوثيقة والتعرف على تسمياتها وتحديد مدى اتساقها وانسجامها مع بعضها البعض. وخلال المقابلات الشخصية مع المستخدمين يقوم محلل الأنظمة بالتعرف على أهمية هذه الوثائق ومدى علاقاتها بعمل المستخدم .

3.3 البرامج الحاسوبية Computer Programs

تستخدم برامج الحاسوب كمصدر للمعلومات في الأنظمة المحوسبة ، فمن خلال هذه البرامج يتم التعرف بشكل تفصيلي إلى بنى البيانات (تراكيب البيانات) المستخدمة في النظام الحالي ، والعمليات التي يجري تنفيذها في النظام.

أما الأسلوب المستخدم للتعامل مع هذا المصدر فهو دراسة البرامج وتحليل الوثائق الخاصة به (وثائق البرامج) .

4.3 أدلة الإجراءات Procedure Manuals :

وتمثل أيضا مصدراً هاماً للمعلومات المتعلقة بالعمليات اليدوية التي يتم أداءها في النظام . حيث تحدد عادة هذه الأدله الأنشطة والمهام التي يقوم بها المستخدم لإنجاز العمليات المختلفة التي يتضمنها النظام .

5.3 التقارير Reports :

تساعد التقارير في تعريف محلل الأنظمة على المخرجات المطلوبة واللازمة لتلبية الإحتياجات المعلوماتية لمستخدمي النظام . وخلال المقابلات الشخصية مع هؤلاء المستخدمين يقوم المحلل بمناقشة محتويات هذه التقارير معهم بهدف مراجعتها

وتنقيحها وتحديد أية متطلبات إضافية لتلبية الإحتياجات الجديدة التي يمكن أن تظهر في المنظمة .

4- الطرق المستخدمة في تجميع المعلومات Search Methods :

يتم إختيار الطرق اللازمة لتجميع المعلومات في ضوء المصادر التي سيتم الحصول منها على هذه المعلومات . وبشكل عام يمكن إختيار واحدة أو أكثر من الطرق التالية لتجميع المعلومات من المصادر المختلفة :

1.4 المقابلات الشخصية Interviews :

وهي من اكثر الطرق استخداماً حيث يتم فيها الحصول على المعلومات من خلال الحوار المباشر بين محلل الأنظمة والمستخدمين ، مما يضمن الحصول على معلومات صحيحة عن عمليات النظام ومشاكله ومتطلبات تطويره .ولكي تكون هذه المقابلات الشخصية ناجحة ومثمرة يجب أن يتم التخطيط لها بعناية ودقة. وفيما يلي بعض العوامل الهامة التي يجب مراعاتها عند استخدام هذه الطريقة لتجميع المعلومات:

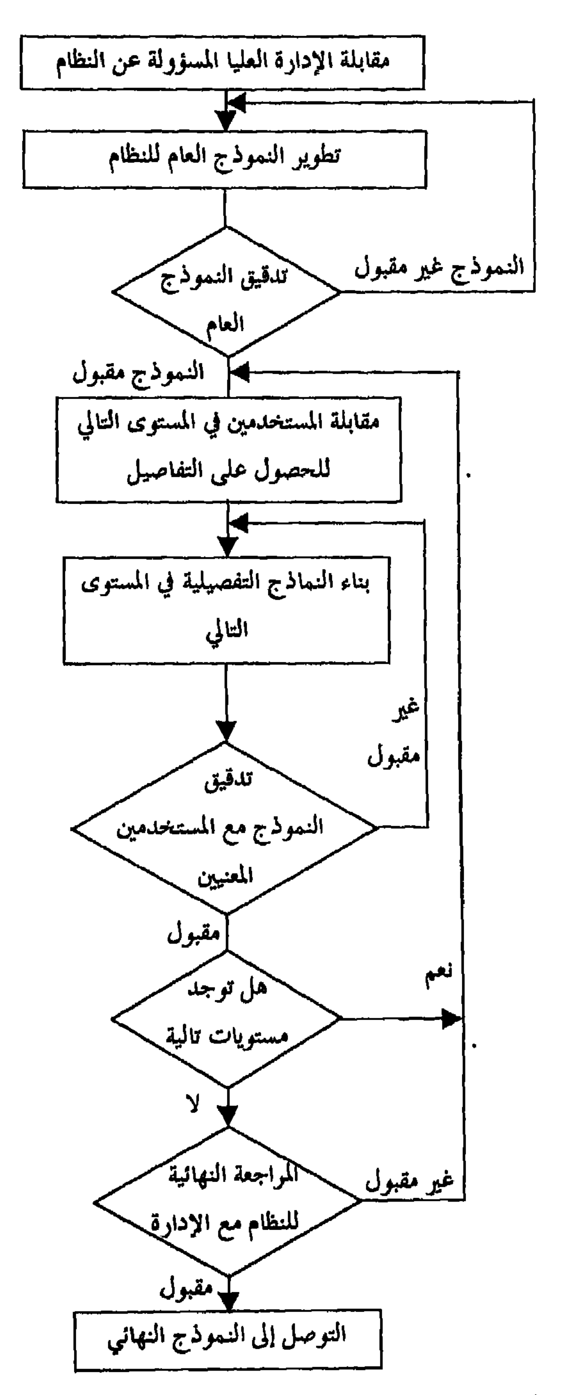
- الاختيار الصحيح للأشخاص الذين ستتم مقابلتهم ، بشكل يضمن الوصول إلى جميع الأفراد الذين لهم علاقة مباشرة مع النظام .
- تحديد التسلسل الصحيح لإجراء المقابلات : حيث يجب البدء بإجراء المقابلات مع المستخدمين الرئيسين Key People للتعرف من خلالهم على الصورة العامة للنظام : أهدافه وعناصره وبيئته والقيود المفروضة على عملياته . ثم تتم متابعة هذه المقابلات مع المستوى الأدنى للحصول على المعلومات التفصيلية . وهكذا فإن الطريقة الصحيحة لإجراء المقابلات تتضمن :

- البدء بمقابلة الإدارة العليا.
- إجراء المقابلات من الأعلى نحو الأسفل كما هو في الشكل (2.4).
- التعرف على التفاصيل المتعلقة بعمليات النظام .
- بناء نموذج النظام (النمذجة) بشكل تدريجي وفقـــاً للتقـــدم في إجراء المقابلات .

• التخطيط للمقابلات من خلال الخطوات التالية :
  - قراءة الوثائق المتعلقة بالنظام .
  - تحديد أهداف المقابلات .
  - تحديد المستخدمين الذين يجب مقابلتهم .
  - إعداد خطة المقابلة لكل مستخدم والاتفاق معه حول موعد المقابلة
  - تحضير الأسئلة التي سيتم طرحها على المستخدم أثناء المقابلة.

• تلخيص نتائج المقابلة بعد الانتهاء منها مباشرة لتوثيـــق المعلومـــات الهامـــة التي تم الحصول عليها .

ويبن الشكل (2.4) تسلسل المقابلات الشخصية مع مستخدمي النظـــام، والذي يجب أن يبدأ بالإدارة العليا ثم يستمر هبوطاً نحو المستويات الأدنى .



شكل (2.3) تسلسل المقابلات الشخصية مع مستخدمي النظام

أما أهم الأسئلة التي يمكن أن يطرحها المحلل أثناء المقابلة فيمكن أن تكـــون على سبيل المثال :

- ما هي المهام التي تقومون بها ؟
- من هم الأفراد الآخرين الذين يمكن أن يقوموا بهذه الأعمال أيضا ؟
- أين ومتى يتم أداء هذه المهام ؟
- ما هي المدخلات (الوثائق أو الأحداث) التي يتم في ضوءها القيـــام بكل مهمة ، وما هي مصادرها ؟
- ما هي المخرجات التي تنتج عن تنفيذ كل مهمة وما هي وجهاتها (لمن توجه هذه المخرجات) ؟
- ما هي الطريقة التي تتبعونها لأداء هذه المهام ؟
- لماذا تقومون بأداء المهام بهذه الطريقة بالذات ؟
- هل هنالك طرق أخرى ممكنة للقيام بهذه المهام ؟
- ما هي اقتراحاتكم للتغير أو التحديث سواء في العمليـــات أو طـــرق التنفيذ ؟

وأخيرا تجدر الإشارة إلى انه في حالة عدم وجود نظام معلومات حـــالي ، أي أن النظام الذي يجري تطويره سيتم استخدامه لأول مـــرة في المنظمـــة ، فـــإن المقابلات في هذه الحالة يجب أن تركز على تحديد متطلبات المستخدمين واللجوء إلى مصادر المعلومات الخارجية ، وكذلك اســـتخدام أســـلوب النمـــاذج التجريبيـــة Prototyping الذي تعرفت إليه في الفصول السابقة.

2.4 الاستبيانات Questionnaires :

يتم تجميع المعلومات وفقاً لهذا الأسلوب من خلال الطلب إلى المستخدمين أن يقوموا بالإجابة على أسئلة مكتوبة في استمارات أو نماذج تصمم خصيصاً لهذا الغرض . ويحتوي نموذج الاستبيان عادةً على مجموعة من الأسئلة المتعلقة بالنظام الحالي ومشاكله ومتطلبات حلها ، ويتم إرساله أو توزيعه على المستخدمين ليقوموا بإملائه وإعادته إلى محلل الأنظمة .

ويلجأ المحلل إلى استخدام أسلوب الاستبيانات في الحالات التي تتطلب الحصول على معلومات من عدد كبير من الأفراد الذين لهم علاقة مباشرة أو غير مباشرة مع النظام . ويكون ذلك غالباً لمعرفة آراءهم وتقييمهم للنظام أو لعناصره، ومدى كفاءة أو فعالية مخرجاته . ويجب أن يراعى عند تصميم هذه الاستبيانات القواعد التالية :

- استخدام الأسئلة المغلقة Closed Questions التي تتطلب اختيار إجابة محددة من بين عدة إجابات ، والإقلال من الأسئلة المفتوحة التي تترك المجال مفتوحاً للمستخدم لكتابة الإجابة التي يراها مناسبة .
- طباعة الاستبيان بأحرف واضحة وترك فراغات كافية بين الأسطر.
- ترك مساحات كافية للإجابة على الأسئلة الموجودة في الاستبيان.
- شرح الهدف من الاستبيان بوضوح لتشجيع الأفراد على الإجابة على أسئلته بموضوعية ودقة.
- المحافظة على التناسق في نمط وأسلوب كتابة أسئلة الاستبيان.
- ترتيب الأسئلة بحيث يبدأ الاستبيان بالأسئلة المهمة للمستخدم والتي يمكن أن تجذب اهتمامه للاستجابة.

- وضع الأسئلة المتعلقة بموضوع أو فقرة ما بجانب بعضـــها في موقـــع واحد .
- وضع الأسئلة المفتوحة والأسئلة التي يتطلب الإجابة عليـــها شـــرح لـــرأي أو وجهة نظر معينة في نهاية الاستبيان .

3.4 تحليل محتويات الوثائق Content Analysis

يستخدم هذا الأسلوب للحصول على المعلومات المتعلقـــة بالنظـــام مـــن المصادر التالية : النماذج – التقارير – الوثائق – أدلة الإجراءات وكذلك الـــبرامج الحاسوبية . فمن خلال هذا الأسلوب يتم تحديد ما يلي :

- تدفقات البيانات ومخازنها ضمن النظام.
- العمليات التي تخضع لها البيانات أثناء حركتها ضمن النظام.
- المخرجات التي يقوم النظام بتوليدها.

وعندما يكون عدد الوثائق المستخدمة في النظام كبيراً يلجأ محلل الأنظمـــة إلى استخدام أسلوب العينات الإحصائية Sampling لاختيار مجموعة محـــددة منـــها والقيام بتحليل محتواها.

4.4 النماذج التجريبية Prototyping :

توفر هذه النماذج طريقة تجريبية تساعد في تحديد متطلبات المســـتخدمين. وتستخدم هذه الطريقة عند تصميم أنظمة جديدة لم يسبق استخدامها، وبالتـــالي لا تتوفر لدى المستخدمين الخبرة الكافية بظروف ومتطلبات عملها . لذلك يقوم محلل الأنظمة بتطوير نموذج تجريبي Prototype للنظام، ويضعه تحت تصرف المســـتخدم الذي يستخدمه بشكل تجريبي لتحديد مدى تلبيته لما هو مطلوب، واقتراح إدخـــال

التعديلات المختلفة فيه . ويتم تطوير هذه النماذج التجريبية باستخدام الأدوات البرمجية الحديثة كلغات الجيل الرابع (Fourth Generation Languages (4GL أو أدوات الحاسوب الشخصي مثل أنظمة إدارة قواعد البيانات أو الجداول الإلكترونية أو لغات البرمجة المرئية ومولدات التقارير ومولدات الشاشات ومولدات التطبيقات وغيرها.

ويمتاز هذا الأسلوب بكونه يشجع المستخدم على المشاركة الفعالة في تحديد المتطلبات المتعلقة بالنظام الذي يجري تطويره.

5.4 الملاحظة المباشرة Observation :

وهي طريقة مباشرة لتجميع المعلومات عن النظام الحالي من خلال المراقبة الفعلية لكيفية سير العمليات فيه والتعرف على مدخلات ومخرجات ومتطلبات تنفيذ هذه العمليات . كما يمكن أن تتم هذه العملية من خلال المشاركة الفعلية لمحلل الأنظمة في تنفيذ العمليات التي يقوم بها النظام ، مما يكون لديه فهماً اعمق وافضل للنظام الذي يجري تطويره. ولكي تكون هذه الطريقة ناجحة يجب التركيز على ملاحظة ما يلي :

- التفاعل بين مستخدمي النظام .
- الأدوار التي يقوم بها المستخدمون المختلفون .
- الأماكن التي يتم فيها تنفيذ هذه الأدوار .
- المهام التي يؤديها كل مستخدم : مدخلات ومخرجات وإجراءات وأدوات تنفيذ كل مهمة .

أسئلة الفصل

1- اشرح أهمية وجود استراتيجية فعالة لتجميع المعلومات اللازمة لتطوير النظام ؟

2- اشرح المكونات الرئيسية لاستراتيجية تجميع المعلومات ؟

3- عدد واشرح مصادر المعلومات التي يمكن من خلالها الحصول على الحقائق والمعلومات اللازمة لدراسة النظام ؟

4- اشرح الطرق المستخدمة في تجميع المعلومات من المصادر المختلفة؟

5- ما هي شروط ومتطلبات المقابلات الشخصية الناجحة ؟

6- اشرح تسلسل المقابلات الشخصية خلال دراسة النظام ؟

7- ما هي الاستبيانات وما هي القواعد التي يجب أن يراعيها محلل الأنظمة عند تصميم الاستبيان ؟

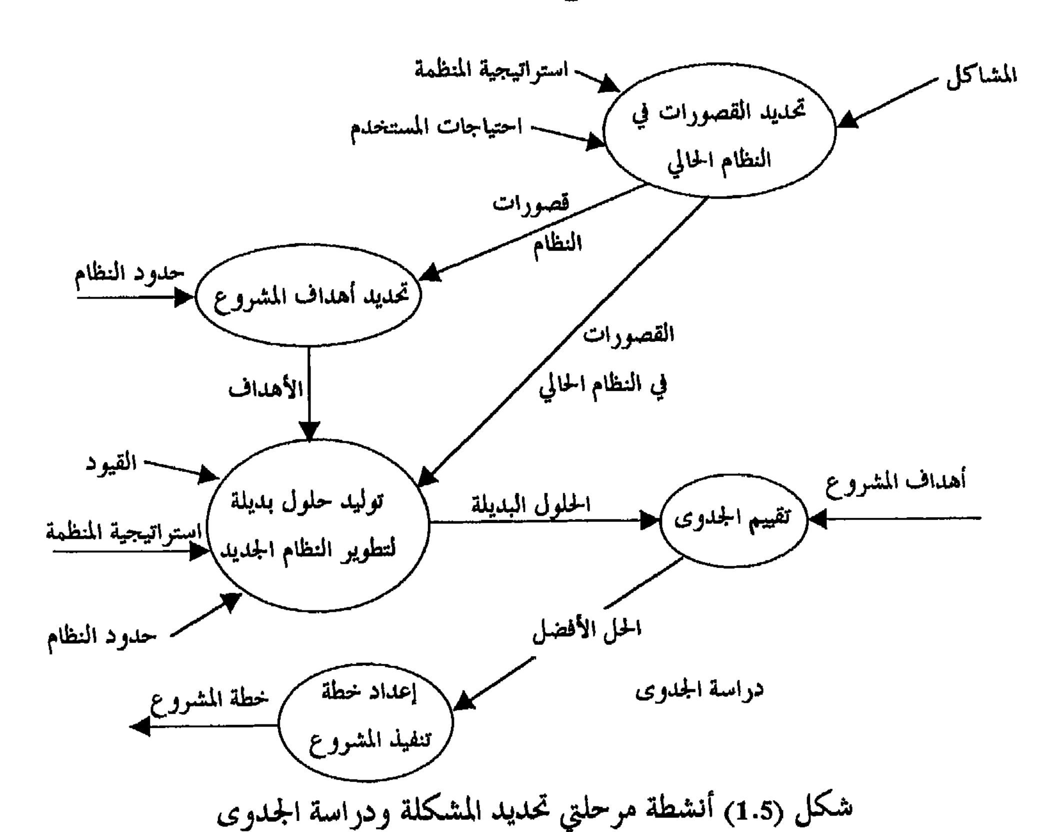
8- ما هي أهمية طريقة الملاحظة المباشرة وما هي متطلبات نجاحها ؟

9- اشرح أهمية النماذج التجريبية كطريقة لتجميع المعلومات عن النظام والحالات التي تستخدم فيها ؟

الفصل الخامس

تحديد المشكلة ودراسة الجدوى

تبدأ مشاريع تطوير أنظمة المعلومات،بمرحلة تحديد أو تعريف المشكلة، حيث يتم تجميع المعلومات اللازمة لفهم النظام الحالي و تحديد المشاكل الموجودة فيه ، والتي دعت إلى التفكير بتغييره أو استبداله بنظام جديد . ثم تنتقل في المرحلة التالية إلى دراسة جدوى الحلول البديلة المقترحة للنظام الجديد واختيار أفضلها ضمن القيود المفروضة والموارد المتاحة لمشروع التطوير.



شكل (1.5) أنشطة مرحلتي تحديد المشكلة ودراسة الجدوى

وسندرس في هذا الفصل بشكل تفصيلي هاتين المرحلتين الهامتين من مراحل تطوير النظام ، ويوضح الشكل (1.5) أهم الأنشطة التي تتضمنها هاتين المرحلتين .

1. مرحلة تحديد المشكلة Problem Definition :

تبدأ هذه المرحلة من الناحية الرسمية بما يسمى " طلب المستخدم " الذي يشعر بوجود مشكلة معينه. وعادة يتضمن هذا الطلب البيانات التالية :

- اسم الإدارة أو القسم الذي توجد فيه المشكلة .
- موضوع المشكلة : مثلاً رقابه المخزن أو محاسبة الزبائن أو غيرها .
- وصف موجز للمشكلة أو الحالة غير المرغوبة .
- ملاحظات المستخدم حول أهمية المشكلة وتأثيرها على سير العمل في المنظمة .

يقوم المحلل في البداية بدراسة طلب المستخدم وتقييم أهمية المشكلة وأولوية حلها من خلال مناقشة النقاط التالية :

- مدى تأثير المشكلة في أهداف المنظمة .
- توفر الخبرات اللازمة لدى إدارة أنظمة المعلومات لحل هذه المشكلة .
- توفر الأجهزة والبرمجيات الحاسوبية اللازمة لحل هذه المشكلة .
- الوفورات المتوقعة في التكلفة أو الفوائد المتوقعة عند حل المشكلة .
- إمكانية تنفيذ الطلب (التوصل إلى حل للمشكلة) ضمن مدى زمني معقول .
- أولوية هذه المشكلة بالنسبة للمشاكل التي تجري معالجتها من قبل إدارة أنظمة المعلومات في الفترة الحالية .

وهكذا يقوم المحلل في هذه المرحلة بتجميع المعلومات التي يمكن أن تساعده في التعرف الدقيق على المشكلة وتحديد عناصرها .

وتتضمن هذه المرحلة نشاطين رئيسيين هما :

1.1 تحديد القصورات أو المشاكل التي يعاني منها النظام الحالي :

يقصد بالمشاكل أو القصورات عدم قدرة النظام على تحقيق الاستراتيجيات المقررة، أو عدم قدرته على تلبية احتياجات المستخدم . وهنا يجب الأخذ بعين الاعتبار أيضاً لحدود النظام الذي تجري دراسته . ويتم تحديد هذه المشاكل أو القصورات بمقارنه العمليات الحالية والأداء الحالي للنظام مع الاستراتيجيات المقررة ومع متطلبات المستخدمين . كما يمكن تحديد القصورات من خلال مقارنه النظام الحالي مع الأنظمة الموجودة في الشركات المماثلة أو في قطاعات الأعمال بشكل ام . وفي النهاية يتم تحديد قائمة بالمشاكل التي يواجهها النظام الحالي والتي يمكن تكون ناتجة عن :

- غياب وظائف هامة لعمل المنظمة (وجود نقص في وظائف النظام الحالي) .
- الأداء غير المقبول للنظام الحالي كبطئ الإجراءات مثلاً ، أو التأخير في استلام ومعالجة الطلبات ، أو تكرار حدوث الأخطاء أو غير ذلك .
- التكلفة الزائدة لعمليات النظام ، بسبب استخدام الطريقة اليدوية المكلفة في عمليات المعالجة، أو بسبب استخدام تقنيات قديمه تتطلب صيانة مكلفة .

- عدم رضى المستخدمين عن النظام الحالي والذي يمكن أن ينعكس في النسب العالية للغياب عن العمل والدوران المرتفع في الموظفين العاملين في النظام .
- شكاوي الزبائن أو الموردين والجهات الخارجية الأخرى ذات العلاقة بعمل النظام . وذلك نتيجة لبطئ الإجراءات ، أو تأخير العمليات مما يؤدي إلى تفويت فرص هامة يمكن أن تستفيد منها المنظمة .

2.1 تحديد أهداف المشروع :

بعد الانتهاء من تحديد المشاكل والقصورات الموجودة في النظام الحالي ، يقوم المحلل بتحديد أهداف مشروع التطوير ، والتي تكون منبثقة بشكل أساسي عن هذه المشاكل والقصورات التي تم تحديدها في الخطوة السابقة . فمثلاً إذا كان النظام الحالي يعاني من عدم وجود بعض الوظائف الضرورية لإدارة المنظمة ، يكون الهدف إضافة هذه الوظائف في النظام الجديد الذي سيتم تطويره . وعندما تكون المشكلة هي الأداء غير المقبول للنظام الحالي يكون الهدف إعادة تصميم عمليات النظام بطريقة جديده تضمن تسريع هذا الأداء وإنجاز العمليات بالكفاءة والفاعلية المطلوبة. وكذلك الأمر عندما تكون المشكلة هي التكلفة الكبيرة، فإن الهدف يكون إعادة تصميم هذه العمليات في النظام الجديد لكي تتم بطريقة أقل تكلفة. وهكذا ففي نهاية هذه الخطوة يتم صياغة أهداف مشروع تطوير نظام المعلومات الحالي والتي يمكن أن تتضمن :

- **تسريع عمليات النظام** (زيادة الإنتاجية) من خلال حوسبة العمليات اليدوية أو استخدام أساليب جديده للحوسبة .

- **تبسيط الإجراءات وترشيد العمليات** من خـــلال التخلـــص مـــن العمليات غير الضرورية .
- **إعادة هندسة العمليات Business Process Reengineering** لتتـــم بطرق جديدة وتقنيات جديدة . وهذا يتضمن أيضا دمج بعض العمليـــات وتقليص خطوات المعالجة إلى أدنى حد ممكـــن لتحســـين كفـــاءة هـــذه العمليات .
- **تقليل الأخطاء** في النظام إلى أدنى حد ممكن من خلال الرقابة والتدقيق على المدخلات للتأكد من صحتها قبل تسجيلها في قواعد البيانات .
- **تحسين واجهة استخدام النظام** لجعلها أكثر سهولة وذلك من خـــلال إعادة النظر في تصاميم (أشكال ومحتويات) المخرجات لإزالة التكرارات الممكنة ، والعمل على تفصيل هذه المخرجات بالتنوع المناســـب لتلبيـــة إحتياجات المستفيدين بفاعلية أعلى.
- **تحسين تكامل النظام** مع الأنظمة الأخرى في المنظمة لضمان تبـــادل البيانات فيما بينها بشكل تلقائي.
- **تحسين رضا الزبائن والموظفين** وغيرهم من الجهات الـــتي يمكـــن أن تتعامل مع نظام المعلومات الذي تجري دراسته.

2– مرحلة دراسة الجدوى Feasibility Study

بعد تحديد أهداف مشروع تطوير نظام المعلومات يجـــب البحـــث عـــن الطـــرق الأكثر جدوى لتحقيق هذه الأهداف . ولذلك فإن هذه المرحلـــة في دورة حياة النظام تركز على إيجاد الطريقة المثلى التي يجب من خلالها تحقيق الأهداف التي

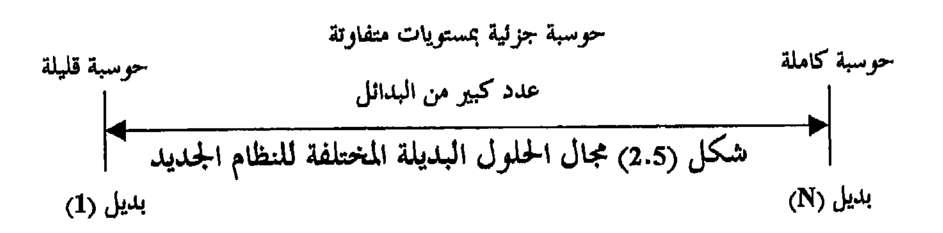
تم تحديدها في المرحلة السابقة ولكي تتم هذه المرحلة بطريقة منهجية فإنها تتضمن ثلاث أنشطة رئيسيه هي :

1.2 توليد الحلول البديلة

في هذه الخطوة الهامة يجب أن يقوم المحلل بصياغة عدد من الحلول البديلة التي يمكن أن تؤدي إلى تحقيق أهداف مشروع التطوير . والحلول التي يتم تطويرها في هذه المرحلة تكون عادة ذات طبيعة عامة أي لا تتضمن تفاصيل كثيرة. والمدخلات التي يجب أن يعتمد عليها المحلل عند توليد هذه الحلول البديلة ، كما هو مبين في الشكل (1.5) هي : الأهداف التي يجب أن تساعد هذه الحلول في تحقيقها، والقيود المتعلقة بالموارد المتاحة للمشروع من تجهيزات وبرمجيات وطاقات حاسوبيه ومخصصات مالية وقيود زمنية وغيرها ، وكذلك الاستراتيجيات التي يجب أن تساعد الحلول البديلة المقترحة في تحقيقها والوصول إليها، وكل ذلك يجب أن يتم في إطار أو ضمن حدود النظام الذي تجري دراسته وتطويره .

ونظراً لأهمية هذه العملية فإنها تتطلب مهارة كبيرة وقدرة على الإبداع والتفكير الخلاق وكذلك القدرة على التخيل والتجريد والتحليل والتركيب .

وبشكل عام فإن الحلول البديلة التي يمكن أن يطورها المحلل في هذه الخطوة يمكن أن تتراوح كما هو مبين في الشكل (2.5) بين أنظمة معلومات محوسبة بشكل كامل Fully Computerized Systems وأنظمة معلومات تتضمن حوسبة عدد قليل من العمليات System with minimal Computerization .



شكل (2.5) مجال الحلول البديلة المختلفة للنظام الجديد

كما يمكن أن تتضمن الحلول البديلة المقترحة للنظـــام الجديـــد خيـــارات أخـــرى مثل :

- الإبقاء على النظام الحالي .
- توظيف عناصر جديدة لتغطية الإختناقات في العمل .
- شراء برمجيات جاهزة .
- تطوير برمجيات خاصة ضمن المنظمة .
- إعادة هندسة العمليات الحالية واستخدام تقنيات وتجهيزات جديدة.

وفي نهاية هذه الخطوة تتكون لدى فريق التطوير مجموعة من الحلول البديلـة لتطوير أنظمة المعلومات ، ولكل حل مزاياه (فوائده) ومتطلباته (تكاليف تنفيذه).

2.2 دراسة الجدوى Feasibility Study

بعد الانتهاء من توليد الحلول البديلة الممكنة لتطوير النظام ، ننتقل في هـــذه الخطوة إلى المفاضلة بين هذه الحلول بهدف إختيار أفضلها. وتمر هذه المفاضلـــة أو التقييم بثلاثة مراحل هي :

1)تقييم الجدوى الفنية Technical Feasibility

وتهدف إلى تحديد فيما إذا كانت التقنية اللازمة لتنفيذ الحل المقترح متاحـة ومتوفرة ويمكن دمجها مع التقنيات الموجودة في المنظمة، والتأكد من توفر الخـــبرات الفنية اللازمة لذلك.

2) الجدوى العملياتية Operational Feasibility

ويقصد بها التأكد من قدرة النظام على توفير المعلومـــات الصحيحـــة في المكان الصحيح وفي التوقيت الصحيح . كما تتضمن الجدوى العملياتية أيضاً التأكد من أن النظام الجديد سيكون مقبولاً ضمن المنظمة ، أي تحديد كيفية ملائمة النظام المقترح للعمليات الحالية في المنظمة ، وفيما إذا كـــانت هنـــاك حاجـــة لإجـــراء تعديـــلات هيكلية فيها .

3)الجدوى الاقتصادية Economic Feasibility :

بعد التأكد من الجدوى الفنية والجدوى العملياتية للحلول البديلة المقترحة للنظام الجديد . يتم دراسة الجدوى الاقتصادية لها بهدف اختيار الحلول التي تحقـــق اكبر فوائد ممكنة بأقل التكاليف، وهذا ما يسمى بتحليـــــل التكلفة والعـــائد أي Cost Benefit Analysis. ولذلك يتم في هذه الخطوة وضع التقديـــرات اللازمـــة لتحديد التكاليف التقديرية والفوائد المتوقعة لكل حل من الحلول البديلـــة الـــتي تم التأكد من جدواها الفنية والعملياتية . وتشمل عملية تقدير التكـــاليف العنـــاصر التالية:

- تكاليف التجهيزات وشبكة الإتصالات .
- تكاليف البرمجيات وقواعد البيانات.
- تكاليف الأفراد وتدريبهم .
- تكاليف المواد من أشرطة ونماذج وغيرها .
- تكاليف التحول من النظام الحالي إلى النظام المقترح .
- أية تكاليف أخرى .

تسمى التكاليف المبينة أعلاه بالتكـــاليف الملموســة Tangible Costs ، لتمييزها عن التكاليف غير الملموسة Intangible Costs التي تكـــون في معظمـــها عبــارة عن تكاليف الفرص الضائعة Lost opportunities والناتجة عن الاســتثمار في نظام معلومات معين بدلاً من نظام آخر .

وبعد الانتهاء من تحديد التكاليف المتوقعة من النظام أو الحلول المقترحة يتـم تحديد الفوائد المتوقعة Estimated Benefits لكل حل، والتي يمكن أيضا أن تكـــون ملموسة (مثلاً التوفير في المواد وأجور القوى العاملة) أو غير ملموسة (مثل تحسين صورة المنظمة لدى الزبائن أو زيادة الرضى الوظيفي للعاملين وغيرها) .

تستخدم التقديرات الخاصة بالتكاليف والفوائد المتوقعة للحلول المقترحـــة للمفاضلة بين هذه الحلول واختيار أفضلها استناداً إلى معايير التقييم المالي التالية :

أ) طريقة فترة الإسترداد Payback Method :

وتحدد الزمن اللازم لكي يستطيع المشروع تغطية تكاليفه أي اســتردادها . والمثال التالي يوضح كيفية استخدام هذه الطريقة :

لنفترض أن تقديرات التكلفة والعائد لأحد المشروعات بـــآلاف الدنانـــير كانت على النحو التالي :

السنة	التكاليف المتوقعة	الفوائد المتوقعة
1	100	-
2	-	20
3	-	40
4	-	60
5	-	40

نجد من البيانات أعلاه أن تكاليف المشروع ستتم في السنة الأولى ومقدارها 100 ألف دينار ، أما الفوائد فستبدأ اعتباراً من السنة الثانية، ويتضح من الجدول أعلاه أن هذا المشروع لن يتمكن من تغطية تكاليفه إلا بعد منتصف السنة الرابعة أي بعد سنتين ونصف من الاستثمار في المشروع. ولكي تكون حسابات فترة الاسترداد أكثر دقة يجب استخدام مفهوم القيمة الحالية الذي سنتعرف عليه في الفقرة التالية.

ب)طريقة القيمة الحالية The Present Value Method :

تستخدم هذه الطريقة لتحديد المبالغ التي يمكن إنفاقها اليوم للحصول على عائد معين في المستقبل . وهذا يعتمد بشكل أساسي على عدد السنوات التي تفصل بين لحظة الاستثمار ولحظة الحصول على العائد المتوقع ، وكذلك نسبة الفائدة التجارية السائدة في السوق . ويتم حساب القيمة الحالية لمبلغ معين حسب العلاقة التالية :

القيمة الحالية = القيمة المتوقعة في السنة N / $(1 + d)^{N-1}$ حيث: d – هي نسبة الحسم التجاري السائدة في السوق، فمثلاً لحساب القيمة الحالية للفائدة المتوقعة في السنة الثالثة للمشروع وهي 40 ألف دينار ، تفصلها عن سنة الاستثمار (وهي السنة الأولى) فترة زمنية مقدارها سنتان ، فإن القيمة الحالية لهذا المبلغ هي :

القيمة الحالية = 40 ألف دينار / $(1 + 0.1)^2$ = 33.5 ألف دينار

أي أن أربعون ألفاً بعد عامين قيمتها الحالية هي فقط ثلاث وثلاثون ألف دينار .

وتستخدم هذه الطريقة لحساب القيمة الحالية الصافية للمشروع Net present value التي تحسب من العلاقة التالية :

القيمة الحالية الصافية للمشروع = القيمة الحالية لفوائد المشروع — القيمة الحالية للتكاليف

والمثال التالي يوضح ذلك :

السنة	التكاليف المتوقعة	الفوائد المتوقعة	(القيم بآلاف الدنانير) القيمة الصافية	
			التكاليف	للفوائد
1	80	—	80	—
2	50	20	45.6	18.2
3	—	40	—	33.6
4	—	60	—	45.6
5	—	80	—	54.8
المجموع	**130**	**200**	**125.6**	**152.2**

وهكذا فإن القيمة الحالية الصافية لهذا الحل هي :

152.2 – 125.6 = 26.6 ألف دينار

وبما أن القيمة الصافية للمشروع موجبة فإنه يعتبر مجدياً من الناحية المالية .

جـ) اختيار الحل الأفضل :

بعد الانتهاء من دراسة الجدوى يتم اختيار الحل الأفضل الذي يكون ذلك الحل الذي يحقق أعلى قيمة صافية أو قصر فترة استرداد .

وعند اختيار هذا الحل يجب الأخذ بعين الاعتبار العوامل التالية :

- العوامل التنظيمية السائدة في المنظمة .
- المهارات الموجودة وإمكانية تنميتها.
- تفضيلات الإدارة بشأن استخدام التقنية المعلوماتية.

كما يمكن اختيار عدة حلول بدلاً من حل واحد لإجراء التحليلات المعمقة لها في المراحل اللاحقة لمشروع تطوير النظام.

3.2– إعداد خطة تنفيذ المشروع

وهي الخطوة الأخيرة في مرحلة دراسة الجدوى حيث يقوم المحلل بوضع خطة عمل مقترحة تتضمن المهام المطلوب إنجازها لتنفيذ المشروع .

يستند تحديد هذه المهام على الحلول المقترحة لتطوير النظام ، وعلى اختيار الأسلوب المناسب للتطوير . ويجب أن توضح خطة العمل التواريخ المقترحة لبدء كل مهمة والانتهاء من إنجازها، وحجم القوى العاملة اللازمة لذلك وما يتطلبه كل نشاط من موارد . أما من حيث الشكل فيمكن أن تكون هذه الخطة بشكل جدول أو قائمة بالمهام ومواعيدها ومتطلباتها، أو بشكل مخطط غانت Gantt chart أو مخطط شبكي أو غير ذلك.

وبعد إنجاز جميع الأنشطة التي تتضمنها دراسة الجدوى يلخص المحلل المعلومات التي توصل إليها في تقرير موجز يسمى **مقترح المشروع** Project Proposal يعرض فيه المزايا التي سيوفرها النظام المقترح لكل من المنظمة والمستخدمين ، كما يجب أن يوضح أهدافه بدقة، وكذلك التحسينات والإضافات التي سيتم إدخالها في النظام الجديد. وبشكل عام يجب أن يقنع هذا التقرير الإدارة بأهمية وضرورة المشروع وجدوى الاستثمار فيه . وبالتالي الموافقة على تنفيذه وتخصيص الموارد اللازمة لذلك . ونظراً لأهمية هذا المقترح فإنه يجب أن يحتوي الفقرات التالية :

- تحديد المشكلة أو المشاكل الموجودة في النظام الحالي وتأثيرها في أداء المنظمة .

- شرح الحل المقترح ومبررات اختياره مع مقارنة موجزة مع بعض الحلول البديلة الأخرى للمشكلة .
- الجدوى الفنية والعملياتية والاقتصادية للحل المقترح .
- الطريقة المقترحة لتنفيذ المشروع ، (أي خطة عمل المشروع المشار إليها في الفقرة السابقة) .
- أدوار الأفراد والإدارات المختلفة التي ستشارك في تنفيذ المشروع .
- التغيرات التي سيحدثها النظام المقترح في طريقة العمل الحالية .وهذا يشمل تحديد الاحتياجات للمهارات الجديدة وطريقة الحصول عليها .

ونظراً لان هذا التقرير موجه بشكل رئيسي للإدارة فانه يجب التأكيد مرة ثانية على أهمية كتابته بعبارات واضحة وبتسلسل منطقي والإقلال ما أمكن من استخدام المصطلحات الفنية.

مثال تطبيقي :

تشرح هذه الحالة المشكلة التي يعاني منها موظفي سجلات المشتركين في مجلة أسبوعية ، والمطلوب قراءة طلب المستخدم وتقرير محلل النظم الذي يتضمن وصفاً أعمق للحالة الراهنة ومقترحات حلها ، ناقش هذه المقترحات وحدد رأيك فيها ؟

طلب المستخدم

إلى : مدير إدارة أنظمة المعلومات

من : رئيس قسم الإشتراكات

الموضوع : نظام متابعة الاشتراكات بالمجلة

منذ تأسيس المجلة نقوم بتنظيم سجلات المشتركين وحفظها بطريقة يدوية في خزانتين للملفات ذات ثمانية أدارج . ولكن حجم العمل الحالي يجعل هذه الطريقة اليدوية غير عملية ومزعجة . نقوم حالياً بمتابعة 75000 مشتركاً مع تجديد اشتراكاهم وتحديث بياناهما (تغير عناوين المشتركين مثلاً) . أصبحت ملفاتنا اليدوية قديمة . بالإضافة إلى ذلك تصلنا العديد من الرسائل من المشتركين يتذمرون فيها من تأخير إرسال المجلات إليهم، وكذلك إرسالها إلى عناوين أخرى بصورة خاطئة . كما تتذمر الإدارة من تقصيرنا بتحصيل رسوم الاشتراكات .

ومما يجعل الأمر اكثر صعوبة . أن إدارة المجلة تطلب منا أحياناً قوائم بأسماء المشتركين وتوزيعاهم حسب المدن والمناطق المختلفة وحسب فئاهم المهنية وأعمارهم . وهذا يعتبر مهمة مستحيلة بالنسبة لنا لقلة العناصر ولما تتطلبه من جهد ووقت كبيرين.

التوقيع

رئيس قسم الإشتراكات

تقرير الدراسة الأولية

إلى : المدير العام

من : محلل الأنظمة

الموضوع : التقرير الأولي لنظام سجلات المشتركين

نلخص فيما يلي ما توصلنا إليه بخصوص مشكلة نظام سجلات المشتركين ونعرض ما نجده حلاً معقولاً . نأمل أن تلبي مقترحاتنا احتياجات الجميع وتجد طريقها إلى التحليل التفصيلي بأسرع ما يمكن، ولقد قمنا بتقدير التكاليف والجدول الزمني اللازم لذلك .

خلال الدراسة الميدانية تبين مايلي :

1- نحتفظ حالياً بسجلات 75000 مشترك تقريباً.

2- يستغرق تسجيل بطاقة الاشتراك زمناً كبيراً (غير عادياً) في المكاتب .

3- يستغرق الموظف 12 أسبوعاً لإعداد التقرير الإداري السنوي.

4- يستلم الموظف 35 طلباً باليوم ويصنفها جميعاً موظف آخر.

5- إن عمل موظف التصنيف صعباً في ظروف المتطلبات الجديدة للإدارة.

6- إمكانية حدوث الأخطاء أو التصنيف الخاطئ أو ضياع الوثائق كبيرة جداً

7- خلال السنة القادمة يجب إضافة ثلاث موظفين آخرين للمحافظة على سير العمل بالطريقة الحالية .

8- يحتاج موظف التصنيف 2-3 دقيقة لاسترجاع ملف المشترك .

9- الخزائن الحالية غير كافية للمحافظة على سير العمل يجب إضافة خزائن

جديدة والمكان صغير لا يتسع لذلك .

استناداً إلى الملاحظات أعلاه اقترح ما يلي :

1- القيام بتحليل تفصيلي بهدف بحث إمكانية حوسبة نظام المشتركين ونظام توزيع المجلة بشكل كامل . يمكن البدء بحوسبة سجلات المشتركين ثم متابعة ذلك ليضم عمليات توزيع المجلة بالبريد على المشاركين .

2- إعادة تصميم بطاقة المشترك لتسهيل استخدامها للإدخال المباشر على الحاسوب .

علماً بأن التحليل التفصيلي يتطلب حوالي 50 ساعة عمل لإنجازه .

التوقيع

محلل الأنظمة

أسئلة الفصل

1- ما هو هدف مرحلة تحديد المشكلة وما هي الأنشطة الرئيسة اللازمة لتنفيذها ؟

2- ما هو المقصود بطلب المستخدم وما هي النقاط التي يجب أن يركز عليها المحلل عند دراسته ؟

3- ما هي الأهداف التي يمكن أن تتضمنها مشاريع تطوير أنظمة المعلومات الحاسوبية ؟

4- ما هو هدف دراسة الجدوى وما هي الأنشطة الرئيسية لها ؟

5- اشرح أهمية توليد اكبر عدد ممكن من الحلول البديلة عند دراسة الجدوى ؟

6- اشرح المراحل الثلاث لتقييم جدوى البدائل المختلفة للنظام (الحلول البديلة) ؟

7- قارن بين طريقة الاسترداد وطريقة القيمة الحالية المستخدمتين في تحديد الجدوى الاقتصادية ؟

8- ما المقصود بخطة عمل المشروع وما هي المعلومات التي تتضمنها ؟

9- ما هي أهم الفقرات التي يجب أن يتألف منها مقترح المشروع ؟

تمرينات

1- تنوي إحدى الشركات استخدام نظام معلومات حاسوبي جديد لرقابة المخزون لديها . وفيما يلي التقديرات المالية الخاصة بتكاليف وفوائد هذا النظام الجديد خلال السنوات القادمة :

السنة	التكاليف	الفوائد
1	3500	15000
2	35000	25000
3	25000	50000
4	45000	65000
5	6000	70000

والمطلوب :

أ) احسب القيمة الحالية للتقديرات الواردة أعلاه بفرض أن نسبة الحسم السنوية هي 8% .

ب) هل المشروع مجدياً من الناحية الاقتصادية ولماذا ؟

2- تنوي إحدى الشركات تطوير نظام معلومات جديد وفيما يلي التقديرات الخاصة لتكلفة وفوائد هذا المشروع بآلاف الدنانير:

السنة	التكاليف	الفوائد
1	85	65
2	60	70
3	51	80
4	75	90

والمطلوب :

أ) ما هي الفترة اللازمة لاسترداد التكاليف المبينة أعلاه .

ب) احسب القيمة الحالية الصافية للمشروع بافتراض نسبة حسم سنوية مقدارها 10% .

جـ) هل يمكن اعتبار المشروع مجدياً من الناحية الاقتصادية ؟ ولماذا .

الفصل السادس

مخططات تدفق البيانات

تعتبر مخططات تدفق البيانات Data Flow Diagrams (DFD) من أهم الأدوات المستخدمة في نمذجة الأنظمة System Modeling. ولقد استخدمت هذه المخططات في العام 1978 لأول مرة في المنهجيات الهيكلية لتحليل الأنظمة Structured System Analysis Methodology، ثم انتشر استخدامها بشكل سريع نظراً لبساطتها وسهولة استخدامها ودقتها في تمثيل الأنظمة ومكوناتها المختلفة.

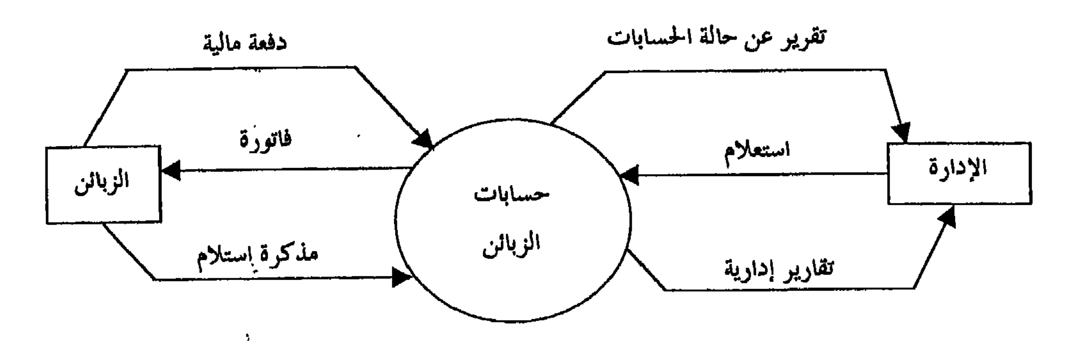
وتهدف المنهجية الهيكلية لتحليل وتصميم الأنظمة إلى تطوير أنظمة معلومات حاسوبية ذات **جودة عالية وسهلة الصيانة وبتكلفة معقولة وخلال فترة زمنية مقبولة**. وتعتمد هذه المنهجية على تقسيم النظام إلى العمليات الرئيسة المكونة له ثم تقسيم العمليات الرئيسة إلى عمليات أصغر وهكذا، حتى الوصول إلى عمليات تفصيلية، في المستوى الأدنى تتضمن كل منها مهمة واحدة محددة فقط.

ويسمى هذا التقسيم بالتحليل او التفكيك الوظيفي Functional Decomposition. وتطبق هذه المنهجية في مرحلة التحليل عند إعداد مخططات تدفق البيانات وبناء نموذج النظام الحالي.

ويدرس هذا الفصل هذا النوع من المخططات وكيفية استخدامها في نمذجة الأنظمة المعلوماتية خلال مرحلتي التحليل والتصميم في دورة حياة تطوير هذه الأنظمة.

1- عناصر مخططات تدفق البيانات

يتم تمثيل مكونات أنظمة المعلومات في مخططات تدفق البيانات كما هو مبين في الشكل (1.6) باستخدام الرموز الأربعة التالية:



شكل (1.6) مخطط تدفق بيانات بيئي (مثال)

أ) **العملية Process** : إن المكون الرئيسي لمخططات التدفق هو **عمليات النظام**، والعملية هي أي عمل أو مهمة أو وظيفة محددة يتم إنجازها لتحويل البيانات الداخلة إليها إلى البيانات الخارجة منها Transformation of Data . يرمز للعملية في مخططات التدفق هذه بشكل دائرة يكتب في داخلها اسم العملية ورقمها.

إن العمليات في مخططات التدفق هي عبارة عن صندوق اسود تعرف مدخلاتها ومخرجاتها فقط، ولكن لا توضح هذه المخططات الخطوات التي يتم بموجبها تحويل البيانات الداخلة إلى المخرجات، فهذه المعلومات نجدها في أداة نمذجة أخرى تسمى **توصيف العمليات** Process Description . سندرسها في فصل لاحق. ولذلك فانه عند إعداد مخططات التدفق يتم التركيز فقط على مدخلات ومخرجات

كل عملية دون الدخول في التفاصيل المتعلقة بما يجري داخلها. كما يجب الانتباه إلى تسمية العملية لتعكس الوظيفة المحددة لها، ويفضل أن تتكون التسمية من كلمتان أو ثلاثة وتكون الكلمة الأولى عبارة عن فعل نشط Active Verb مثلاً احسب الضريبة، أدخل العنوان، احصل على السعر وهكذا. ولذلك فان التسميات العامة أو غير الواضحة مثل "عالج السجل" أو " عدل العنوان" يجب تجنب استخدامها.

ب – مخزن البيانات Data Store

يستخدم هذا العنصر لتمثيل مخازن البيانات الموجودة في النظام، ومخزن البيانات هو عبارة عن مستودع أو مكان تستقر فيه البيانات بشكل دائم أو مؤقت حتى تظهر الحاجة إليها من قبل عمليات النظام، ويمكن أن تكون هذه المخازن عبارة عن ملفات يدوية، أي مجموعة بطاقات ووثائق، أو بشكل أشرطة أو أقراص ممغنطة أو غيرها.

يرمز لمخازن البيانات في مخططات تدفق البيانات بشكل مستطيل مفتوح من أحد جوانبه ويكتب فيه اسم المخزن. على النحو التالي:

سجلات المبيعات

تسمى مخازن البيانات عادة باسم الجمع للسجلات الموجودة فيها فمثلاً سجلات الزبائن أو سجلات المواد أو غيرها، ولا تتم في مخازن البيانات أية عمليات تحويل بل تكون البيانات الخارجة منها هي نفس البيانات المدخلة إليها، أما اتجاهات الأسهم فتبين حركة البيانات منها واليها، وتقوم العمليات Processes بإدخال

البيانات إلى مخازن البيانات واسترجاعها منها، وأخيراً تجدر الإشارة إلى أن مخازن البيانات لا تظهر في مخططات التدفق البيئية لكونها من العناصر الداخلية للنظام.

جـ) تدفق البيانات Data Flow:

يستخدم هذا العنصر لتمثيل حركة البيانات داخل النظام (أي بين عمليات النظام المختلفة، وكذلك لتمثيل حركة البيانات بين عناصر البيئة الخارجية وعمليات النظام.

ويرمز لتدفق البيانات بشكل سهم يكتب فوقه اسمه، وتدفق البيانات هو عبارة عن حزمة Package أو دفعة Batch، أو شحنة من البيانات يتم إرسالها من عملية إلى أخرى أو يتم استلامها من خارج النظام، أو إرسالها من قبل عملية معينة إلى عناصر خارج النظام. ولذلك فان تدفقات البيانات تمثل حركة البيانات في النظام، ويمكن أن تتكون من حقل بيانات واحد أو من سجل مؤلف من عدة حقول مرتبطة معاً ومتماسكة منطقياً، أما إذا كانت هذه الحقول غير متماسكة منطقياً فيتم تمثيلها بعدة أسهم، ويجب أن يكون لكل تدفق بيانات اسماً فريداً ذو دلالة واضحة فمثلا التسمية READ-ERROR-MESSAGE أكثر وضوحاً ودلالة من التسمية MESSAGE1.

تستخدم أسماء التدفقات بصيغ المفرد، وهذا يعني أن مخططات تدفق البيانات لا تبين التكرارات أي الدورات Loops في عمليات المعالجة بل تركز على حركة البيانات.

د) الكينونة الخارجية External Entity:

يستخدم هذا العنصر في مخططات تدفق البيانات لتمثيل الكينونات الخارجية للنظام، أي عناصر البيئة الخارجية للنظام التي تتبادل البيانات معه،

فالنظام يمكن أن يحصل على البيانات من هذه العناصر أو يزودها بالمعلومات اللازمة لها، ففي نظام المبيعات يكون الزبائن عبارة عن كينونة خارجية يتلقى منهم طلبات المبيعات ويرسل إليهم البيانات المتعلقة بتسليم المنتجات أو الخدمات المطلوبة والفواتير الخاصة بهذه المبيعات.

يرمز للكينونات الخارجية هذه، والتي تسمى أيضا Terminators في مخططات تدفق البيانات بشكل مستطيل يكتب في داخله اسم الكينونة. وبشكل عام فان الكينونة الخارجية هي شخص أو جهة أو منظمة تقع خارج حدود النظام الذي تجري دراسته، ويكون لها علاقة مباشرة مع النظام، إما بتوريد البيانات إليه أو أن النظام يورد لها بيانات.

2- مستويات مخطط تدفق البيانات

تستخدم مخططات تدفق البيانات لوصف نظام المعلومات الذي تجري دراسته، وتسمى هذه العملية كما ذكرنا نمذجة عمليات النظام، أي أنه يتم تمثيل النظام بشكل تجريدي باستخدام رموز هذه المخططات. وتتضمن منهجية التحليل الهيكلية Structured Analysis تقسيم النظام من الأعلى إلى الأسفل إلى مكوناته الوظيفية، ولذلك فانه يتم إعداد مخططات تدفق البيانات في عدة مستويات هي:

- **مخطط تدفق البيانات البيئي Context DFD** الذي يعكس بيئة النظام فقط، أي علاقة النظام بالعناصر الخارجية في البيئة المحيطة، وهذا يعني أن هذا المستوى من المخططات يحدد هذه العناصر الخارجية وتدفقات البيانات بينها وبين النظام. ولذلك يمثل النظام في هذه المخططات بشكل دائرة واحدة كما هو مبين في الشكل (1.6).

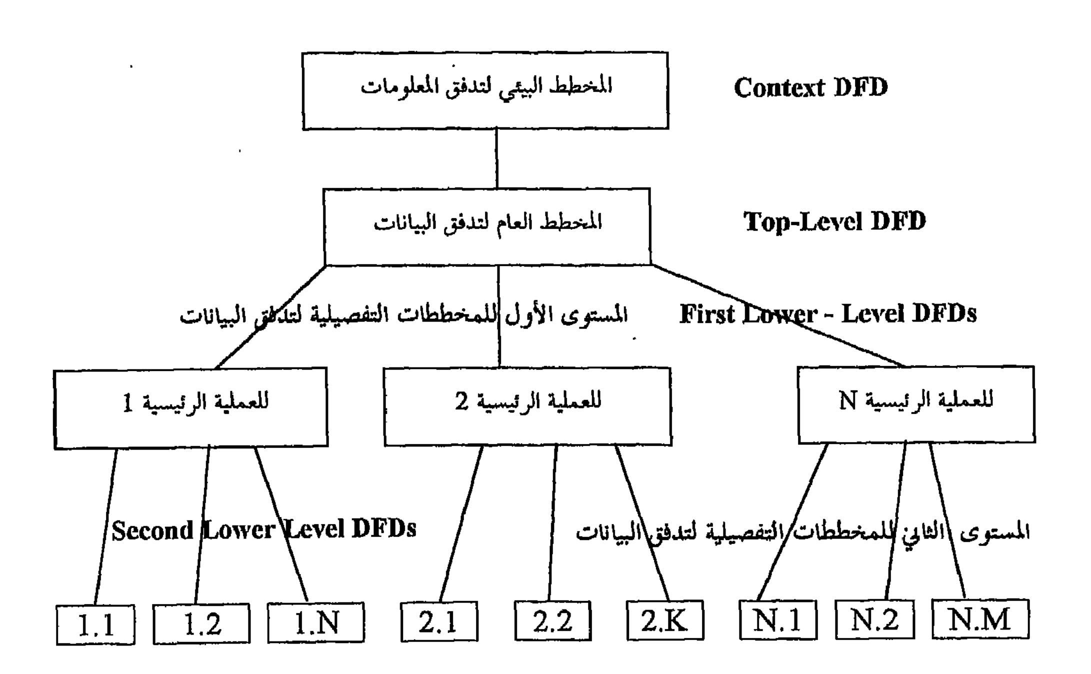
- المخطط العام أو مخطط المستوى الأعلى لتدفق البيانات:

ويسمى Top-Level DFD أو Overview DFD ويعكس العمليات أو الوظائف الرئيسة للنظام، حيث يتم تقسيم النظام إلى مجموعة من العمليات الرئيسة وتمثيلها في المخطط ورسم تدفقات البيانات فيما بينها.

- المخططات التفصيلية أو مخططات المستويات الدنيا:

حيث يتم تمثيل كل عملية من العمليات الرئيسة الموجودة في المخطط العام بشكل مخطط تفصيلي يبين مكونات هذه العملية الرئيسة من عمليات تفصيلية وتدفقات ومخازن بيانات وهكذا، ويمكن أن تكون هذه المخططات التفصيلية في عدة مستويات، أي يمكن أن يكون لكل عملية في المخطط التفصيلي الأول مخطط تفصيلي في المستوى الأدنى.

تسمى المخططات التفصيلية للعمليات الرئيسة First Lower-Level DFD، فإذا تطلب الأمر إعداد مخطط تفصيلي لبعض العمليات الموجودة في هذا المخطط، يتم إنشاء مستوى أدنى جديد وتسميته بالمستوى التفصيلي الثاني Second lower-Level DFD وهكذا. ويبين الشكل (2.6) هيكلية مخططات تدفق البيانات:



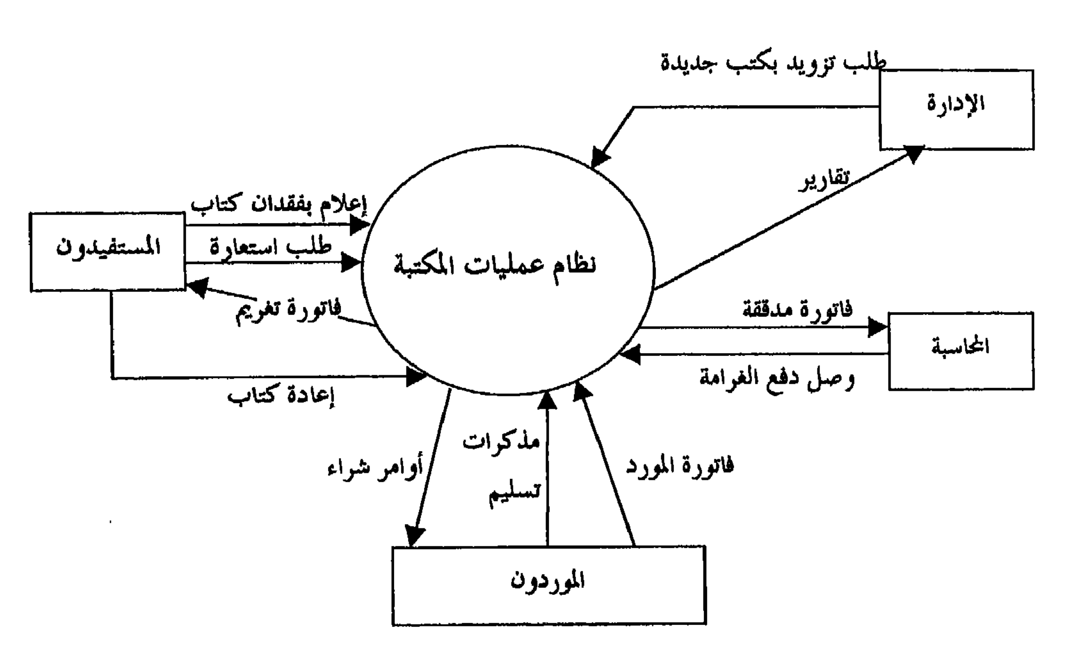
شكل (2.6) هيكلية مخططات تدفق بيانات النظام

1.2 المخطط البيئي لتدفق البيانات Context DFD

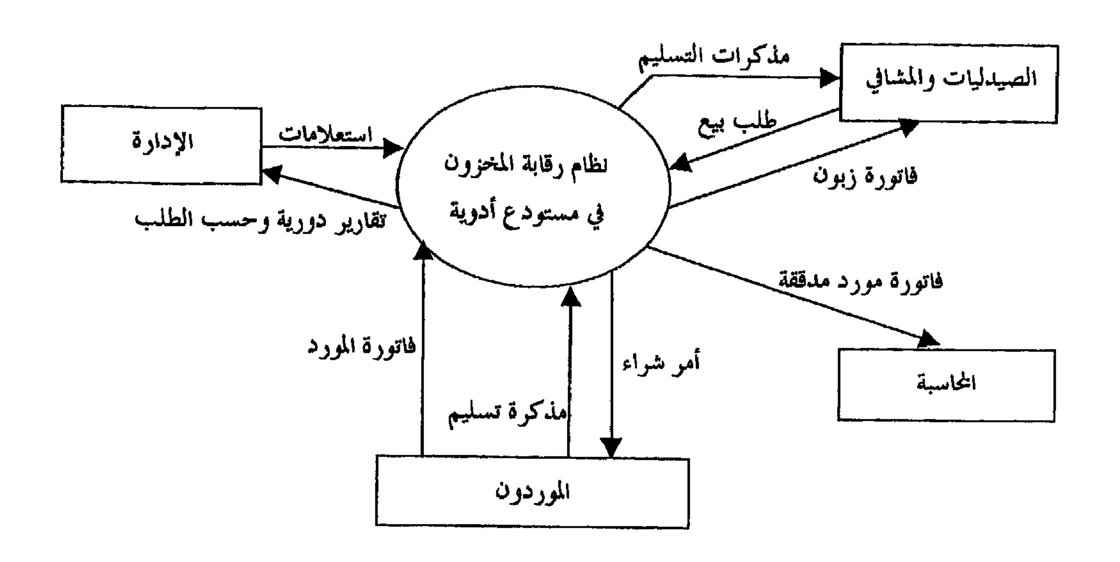
يأتي هذا المخطط في أعلى مستوى في مخططات التدفق، حيث يمثل النظرة الإجمالية العامة للنظام، ويركز على تبيان علاقة النظام مع عناصر البيئة المحيطة به ولذلك كانت تسميته بالمخطط البيئي. تستخدم في رسم هذا المخطط العناصر التي تعرفنا إليها في الفقرة السابقة باستثناء مخازن البيانات، فهي عنصر داخلي لذلك لا تظهر هذه المخازن في المخططات البيئية، وبشكل عام فان المخطط البيئي لتدفق البيانات يمثل أو يعكس ما يلي (انظر الأشكال (3.6) و (4.6)):

- **الوظيفة العامة للنظام:** ويتم تمثيلها بشكل دائرة يكتب في داخلها اسم النظام، والمخططات البيئية لا تحوي سوى عملية واحدة تمثل النظام بأكمله.
- **الكينونات الخارجية للنظام:** أي عناصر البيئة الخارجية التي يمكن أن تكون مصدراً للبيانات اللازمة للنظام أو التي يمكن أن تكون وجهة لمخرجات النظام أي تستلم البيانات من النظام.
- **تدفقات البيانات بين النظام وعناصر البيئة الخارجية.**

وبشكل عام فان هذا المخطط يهدف إلى تبيان حدود النظام الذي تجري دراسته، وتبين الأشكال (3.6) و (4.6) أمثلة عن المخططات البيئية لتدفق البيانات لأنظمة مختلفة.



شكل (3.6) المخطط البيئي لتدفق البيانات لنظام المكتبة



شكل (4.6) المخطط البيئي لتدفق البيانات لنظام رقابة المخزون في مستودع توزيع أدوية

ويتضح من الأشكال (3.6) و (4.6) أن المخطط البيئي يركز على توضيح حدود النظام وتبيان عناصر البيئة الخارجية (الكينونات الخارجية) وعلاقتها مع النظام، فمثلا نجد أن عناصر البيئة الخاصة بنظام المكتبة هي المستعيرون والموردون والمحاسبة والإدارة، وترتبط هذه الكينونات مع النظام من خلال تدفقات البيانات فيما بينها وبين النظام، والتي تتمثل في المخطط بشكل أسهم تكتب بجانبها أو فوقها أسماء هذه التدفقات. وفي نظام رقابة المخزون في مستودع الأدوية يتقدم الزبائن (الصيدليات والمشافي) بطلبات البيع إلى النظام الذي يقوم بمعالجة هذه الطلبات وإعداد مذكرات التسليم والفاتورة المتعلقة بالطلبية لإرسالها إلى الزبون صاحب الطلب. كما يقوم النظام بإرسال أوامر الشراء إلى الموردين ويستقبل منهم مذكرات التسليم والفواتير المتعلقة بالشحنات المسلمة بناء على أوامر الشراء هذه. ويقوم

النظام بتدقيق هذه الفواتير وإرسالها إلى المحاسبة لكي يتم العمل على دفعها في الوقت المناسب.

أما الإدارة فتتعامل مع النظام من خلال حصولها على التقارير التي يمكن أن يولدها بصورة دورية أو حسب الطلب، كما يمكن للإدارة القيام بالاستعلامات المختلفة للحصول على المعلومات اللازمة لاتخاذ القرار.

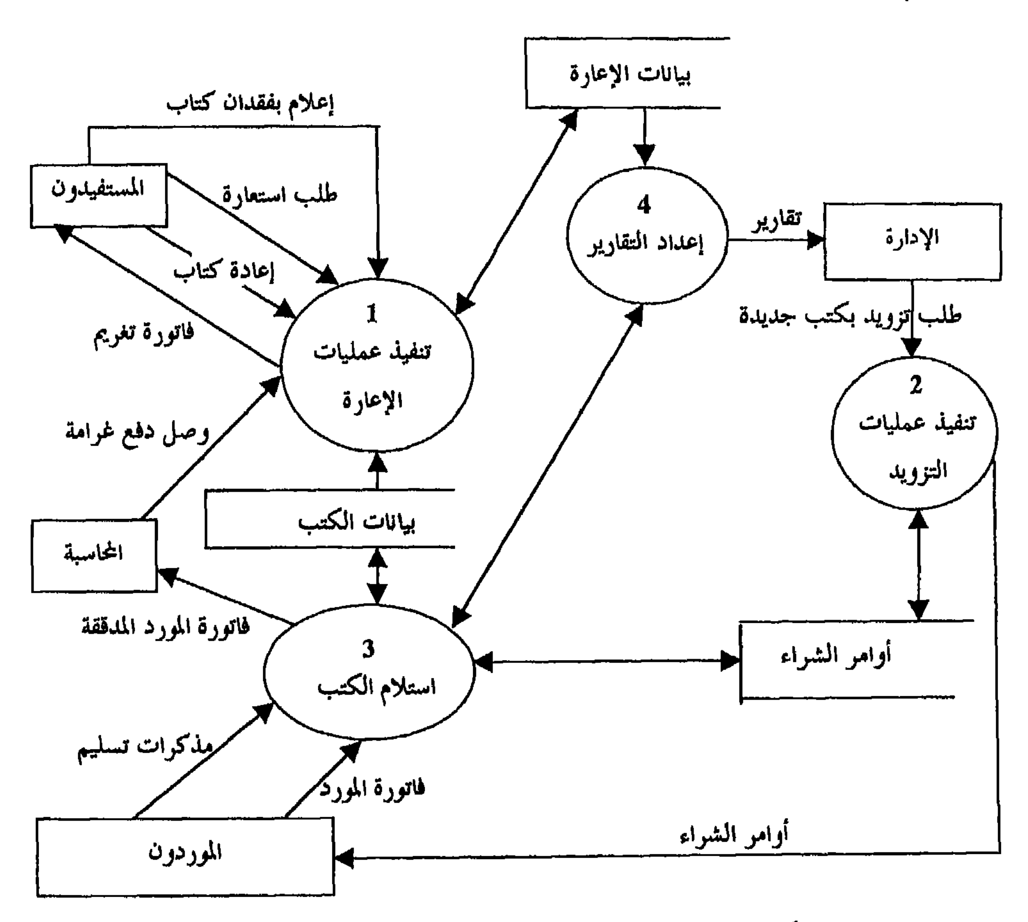
2.2 المخطط العام لتدفق البيانات Overview DFD:

ويسمى هذا المخطط أيضا بمخطط المستوى الأعلى لتدفق البيانات Top-Level DFD ويتم في هذه المخططات إعطاء تفاصيل اكثر عن النظام، حيث يتم تقسيمه إلى مجموعة من العمليات الرئيسة وتبيان علاقتها ببعضها البعض من خلال تدفقات البيانات فيما بينها. كما يمكن أن تظهر في هذا المستوى مخازن البيانات إذا كانت تستخدم من قبل اكثر من عملية واحدة. وهكذا فان هذا المستوى من مخططات التدفق يركز على تبيان العمليات (الوظائف) الرئيسة للنظام الذي تجري دراسته، ولذلك سمي بالمخطط العام لتمييزه عن المخططات التفصيلية التي سيتم إعدادها لكل عملية من هذه العمليات الرئيسة، ويبين الشكل (5.6) لتدفق البيانات. ويتضح من هذا الشكل أعلاه أن نظام المكتبة المبين مخططه البيئي في الشكل (2.6)، يحتوي على أربعة عمليات رئيسة تم ترقيمها وتسميتها على النحو التالي:

1- إعارة الكتب
2- التزويد بالكتب (شراء الكتب)
3- استلام الكتب

4- إعداد التقارير الإدارية

كما يبن المخطط العام في الشكل (5.6) أن النظام يستخدم في هذا المستوى ثلاث مخازن بيانات مشتركة بين العمليات وهي: بيانات الكتب وبيانات الإعارة وبيانات أوامر شراء الكتب. وأخيرا يبين المخطط العام لتدفق البيانات في نظام المكتبة تدفقات البيانات من الكينونات الخارجية إلى هذه العمليات الرئيسة الأربعة وبالعكس، وكذلك تدفقات البيانات بين هذه العمليات، وبينها وبين مخازن البيانات المختلفة.

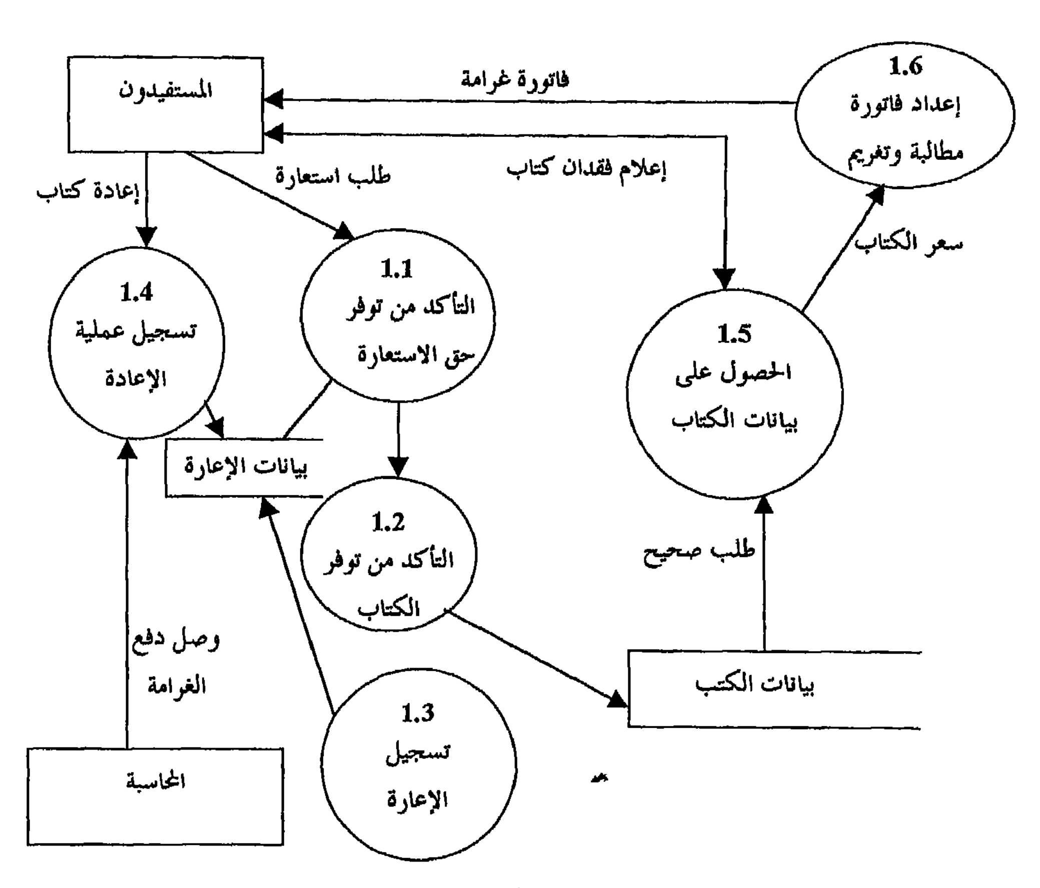


شكل (5.6) المخطط العام لتدفق البيانات في نظام المكتبة

3.2 المخططات التفصيلية أو مخططات المستوى الأدنى لتدفق البيانات

يتضمن هذا المستوى من مخططات تدفق البيانات مخططات منفصلة لكل عملية من العمليات المبينة في مخططات التدفق في المستوى الأعلى. فمثلا يتضمن المخطط العام المبين في الشكل (5.6) أربعة عمليات رئيسة، وبالتالي يجب إعداد أربعة مخططات منفصلة يمثل كل منها العمليات التفصيلية المكونة لإحدى هذه العمليات الرئيسة، وتسمى هذه المخططات في هذا المستوى الذي يأتي بعد المخطط العام لتدفق البيانات في النظام **بمخطط المستوى التفصيلي الأول** أو **مخطط المستوى الأدنى الأول لتدفق البيانات** First Lower-Level DFD. وفي هذه المخططات يتم أيضا تقسيم كل عملية رئيسة إلى عدد من العمليات الأصغر وتحديد تدفقات البيانات فيما بينها، وكذلك مخازن البيانات التي تستخدمها، ويبين الشكل (6.6) مثالا للمخطط التفصيلي الأول (المستوى الأدنى الأول) لتدفق البيانات الخاص بالعملية رقم 1 المتعلقة بإعارة الكتب، ويتضح من هذا المخطط أن عملية إعارة الكتب تتكون من ستة عمليات فرعية هي:

1- التأكد من توفر حق الإعارة.
2- التأكد من توفر الكتاب المطلوب.
3- تسليم الكتاب وتسجيل بيانات الإعارة.
4- تسجيل عملية إنهاء الإعارة عند إعادة الكتاب.
5- الحصول على سجل الكتاب المفقود من قبل المستعير.
6- إعداد فاتورة غرامة وإرسالها إلى المستعير.



شكل (6.6) المخطط التفصيلي الأول لتدفق البيانات للعملية رقم 1
"إعارة الكتب في نظام المكتبة"

كما يبين هذا المخطط تدفق البيانات بين هذه العمليات ومخازن البيانات والكينونات الخارجية في النظام. كما نلاحظ أن ترقيم العمليات يبدأ بالرقم 1 وهو رقم العملية الرئيسة في المخطط العام، يليه نقطة ثم رقم العملية الفرعية في هذا المخطط التفصيلي.

وأخيراً تجدر الإشارة إلى إمكانية المضي قدماً نحو الأسفل وإعداد المخططات التفصيلية للمستوى التالي لكل عملية من العمليات الموجودة في المخطط التفصيلي

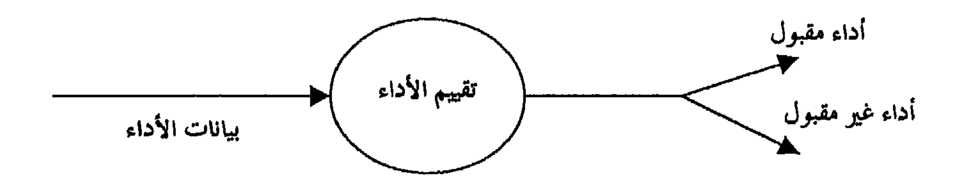
الأول. وهكذا يمكن الاستمرار حتى تصبح كل عملية تمثل وظيفة محددة واحدة فقط، أو يمكن وصفها (توثيقها) باستخدام اللغات البنيوية في أقل من صفحة واحدة.

3. خصائص مخططات تدفق البيانات DFD Features

يجب أن تكون مخططات تدفق البيانات سهلة الفهم Self-Explanatory وواضحة لا إبهام فيها Unambiguous، وكاملة تعكس جميع عمليات النظام وعناصره المختلفة، ولكي تحقق مخططات تدفق البيانات هذه المتطلبات يجب أن يتوفر فيها الخصائص التالية:

1.3 أن لا تحتوي التراكيب أو البنى المستخدمة في مخططات تدفق البرامج Flowcharts: وهذا يعني أن مخطط تدفق البيانات الجيد يجب أن لا يتضمن:

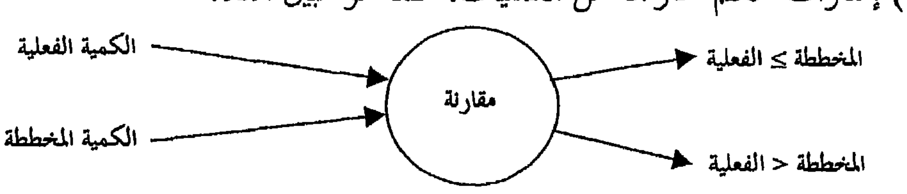
أ) تدفقات بيانات يتم تقسيمها إلى عدة تدفقات كما هو مبين أدناه:



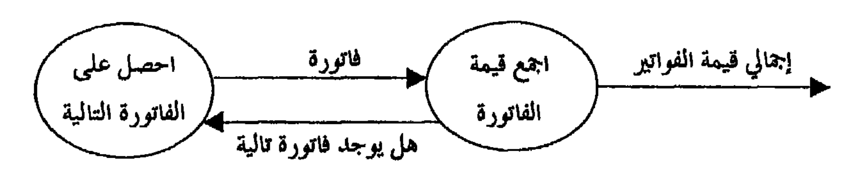
ب) تدفقات بيانات (أسهم) متقاطعة.

ج) تدفقات بيانات بين الكينونات الخارجية، فأي تدفق يجب أن يصدر من عملية أو يصب فيها.

د) إشارات تحكم خارجه من العمليات، كما هو مبين أدناه:

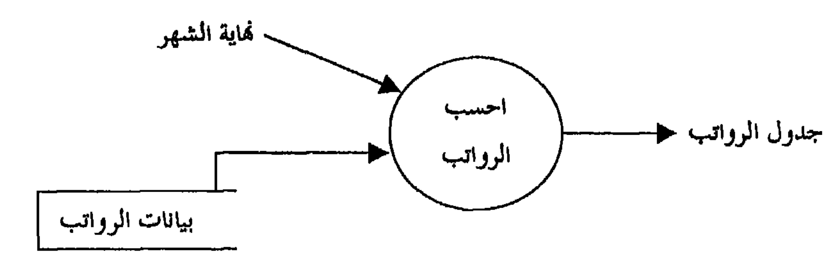


هـ) دورات أو تكرارات تتضمن عناصر تحكم، مثلا.



في المثال أعلاه لا يمثل السهم " هل يوجد فواتير أخرى" تدفق بيانات بل عنصر تحكم Control Element ولذلك فانه غير مقبول في مخططات تدفق البيانات.

و) إشارات إدخال Input Signals كما هو مبين أدناه:



وهنا أيضا فان تدفق البيانات المسمى "نهاية الشهر" هو عبارة عن إشارة تحكم وليس تدفق بيانات.

ز) أكثر من تدفق بيانات واحد بين نفس العمليتين.

ح) تدفق بيانات مباشر من الكينونات الخارجية إلى مخازن البيانات الموجودة في النظام.

2.3 التقيد بمبدأ حفظ البيانات:

يقصد بحفظ البيانات Conservation of Data أن البيانات لا تخلـــق مـــن عدم ولا تفنى في النظام. أي أن النظام لا يمكنه خلق البيانات من لا شيء بل مـــن بيانات موجودة فيه أو واردة إليه، وكذلك فان البيانات الداخلة إلى النظام لا يمكــن أن تفــنى وتتلاشى بل تستقر في مخازن البيانات أو تخرج إلى خارج النظام.

وينطبق هذا المبدأ على كل من العمليات ومخازن البيانات علـــى الســـواء ففـــي مخازن البيانات لا يمكن أن تخرج من هذه المخازن بيانات لم ترد إليها سابقا، أما بالنسبة للعمليات فإنها لا يمكن أن تخلق بيانات جديدة، وإنما تقـــوم باســـتلام البيانات الداخلة إليها وتحويلها إلى شكل جديد ثم إخراجها وفـــق هـــذا الشـــكل الجديد، كما أنه لا يمكن أن تضيع في العملية أية بيانات.

3.3 استخدام تسميات واضحة ذات دلالة:

تساعد التسميات الجيدة في تسهيل فهم مخططات التدفق وإيضاح محتوياتها، ولذلك يجب التقيد بالملاحظات التالية عند رسم المخططات وتســـمية عناصرهـــا المختلفة:

أ) **تسمية العمليات:** يجب استخدام عبارة صغيرة واحدة كافية لوصف العملية، ويجب أن تعكس هذه العبارة عمـــل أو وظيفـــة محـــددة Specific Action مثل " احسب الضريبة" أو " اجمع قيمة الفـــاتورة إلى الإجمالي اليومي" أو غير ذلك. ولذلك يجب تجنـــب اســـتخدام التسميات العامة مثل " افحص الفاتورة"

ب) **تسمية مخازن البيانات:** يجب استخدام أسماء واضحة ومحددة والابتعاد عن التعابير العامة كما يجب أن يحتوي مخزن البيانات مجموعة واحدة من التراكيب، فلا يمكن مثلا استخدام مخزن بيانات واحد للاحتفاظ بالفواتير وأوامر الشراء.

جـ) **تسمية تدفقات البيانات:** يفضل استخدام كلمة واحدة لتسمية هذه التدفقات مثل: فاتورة أو طلبية أو إيصال أو غيرها. ولكنه غالبا نحتاج إلى استخدام كلمتان أو اكثر مثل " أمر شراء" أو " إيصال دفع" أو "فاتورة المورد" أو غير ذلك، وهنا أيضا يجب التذكير بضرورة استخدام تسميات واضحة ومحددة وتعبر بدقة عن هذه التدفقات.

4.3 الربط الجيد للمستويات المختلفة لمخططات تدفق بيانات النظام.

عرفنا مما سبق أن مخططات تدفق البيانات يتم إعدادها في عدة مستويات ولضمان صحة هذه المخططات وربطها معا وعدم وجود اختلاف في محتوياتها يجب استخدام الأساليب التالية:

أ) **ترقيم العمليات الموجودة في المخططات** حيث تعطى العملية الوحيدة في المخطط البيئي الرقم صفر، ويسمى هذا المخطط أيضا Zero-Level DFD. أما في المخطط العام فتعطى العمليات أرقام متصاعدة أي 1،2،3،... وهكذا. ثم عند الانتقال إلى المخططات التفصيلية لهذه العمليات يتم ترقيم هذه العمليات باستخدام رقمين يفصل بينهما نقطة ويكون الرقم الموجود إلى يسار النقطة هو رقم العملية الرئيسة، كما هو مبين في الشكل (6.6).

ب) **موازنة تدفقات البيانات Balancing of data flows** : ويقصد بها التأكد من أن جميع تدفقات البيانات الداخلة إلى عملية معينة هي نفس التدفقات الداخلة إلى المخطط التفصيلي لتدفق البيانات الخاص بهذه العملية، وكذلك الأمر بالنسبة للتدفقات الخارجة من العملية.

إن التقيد بمتطلبات موازنة التدفقات هذه تجعل المخططات في المستويات العليا تحتوي على عدد كبير من التدفقات مما يجعل هذه المخططات معقدة ومتشابكة. ولمعالجة هذه المشكلة يتم اللجوء إلى تمثيل تدفقات البيانات المتشابهة في المخططات الدنيا بتدفق واحد في المخطط الأعلى، كما يمكن إهمال رسم بعض التدفقات غير الهامة أو ذات الطبيعة البسيطة (مثل الحوارات Dialogues بين العمليات ومعالجة الأخطاء وغيرها) في المخططات ذات المستوى الأعلى، وتسمى هذه العملية Expanding of data flows.

5.3 حدود التقسيم الأفقي والعمودي للعمليات أثناء إعداد مخططات التدفق:

لكي تكون مخططات التدفق بجميع مستوياتها ذات بنية واضحة ومفهومة جيدا فانه ينصح بما يلي:

أ) الاستمرار في التقسيم إلى المستوى الأدنى حتى الوصول إلى عمليات تتضمن **وظيفة أو مهمة محددة واحدة**، أو يمكن توصيفها بأقل من صفحة واحدة.

ب) عند تقسيم كل عملية إلى العمليات المكونة لها في المستوى الأدنى فان عدد هذه العمليات الفرعية يفضل أن يكون بين 5 إلى 9 عمليات.

ج) يجب الحرص عند تقسيم كل عملية إلى العمليات المكونة لهـــا أن تكون التدفقات بين هذه العمليات أقل ما يمكـــن لكـــي يكـــون المخطط أكثر وضوحا.

6.3 متطلبات أخرى:

يجب كتابة أسماء التدفقات وكذلك أسماء العمليـــات ومخـــازن البيانـــات بحـــروف كبيرة سواء في المخططـــات أو في قـــاموس البيانـــات أو في توصيـــف العمليات.

كما تجدر الإشارة إلى إمكانية رسم مخازن البيانات وكذلـــك الكينونـــات الخارجية في أكثر من مكان في المخطط، وذلك بهدف تبســـيطه والإقـــلال قـــدر الإمكان من الخطوط المتقاطعة فيه، وفي هذه الحالة يتم وضع خط إضافي في رمـــوز هذه العناصر عند تكرار رسمها في نفس المخطط وذلك على النحو التالي:

الزبائن

بيانات الزبائن

4. المخططات المادية والمنطقية لتدفق البيانات:

تعكس المخططات المادية Physical DFD كيفية عمل النظام أي العمليات التي يقوم بها والطريقة التي يتم بواسطتها أداء هذه العمليات. وعندما يبـــدأ المحلـــل بتجميع المعلومات عن النظام الذي تجري دراسته يقوم أولا برسم المخططات المادية لتدفق البيانات، ثم بعد ذلك يقوم بتحويل هذه المخططات الماديـــة إلى مخططـــات منطقية Logical DFD تمثل فقط العمليات التي يتم أداءها دون أي إشارة للطريقة أو

الكيفية التي تتم فيها هذه العمليات. وذلك لكي يستطيع المحلل التركيز على ما هـو مطلوب أداءه من قبل النظام، والابتعاد عن التفاصيل المتعلقة بكيفية التنفيذ. وتتـم عملية التحويل هذه وفق الخطوات التالية:

1) تحذف من المخطط المادي جميع العمليات المادية التي لا يتم خلالها إجراء أي تحويل للبيانات.
2) تستبدل العمليات المادية المتبقية في المخطط والـتي يتـم خلالها عمليات تحويل للبيانات، بعملية أو عمليـات تعكـس الوظيفـة المنطقية لها، ويمكن أن تستبدل العملية الماديـة الواحـدة بعـدة عمليات منطقية.
3) يفحص المخطط الناتج ويتم تجميع العمليات المتشابهة معا ويرسـم مخطط التدفق الجديد والذي يكون بمثابة مخطط منطقي يحدد فقـط العمليات التي يتم أدائها في النظام.

5. خطوات إعداد مخططات تدفق البيانات:

لرسم مخططات تدفق البيانات يمكن اتباع الخطوات التالية:

1) **تحديد الكينونات الخارجية للنظام وتدفقات البيانات التي تربطها بـه**، أي مدخلات ومخرجات النظام.
2) **رسم المخطط البيئي لتدفق البيانات** في ضوء المعلومات التي تم تحديدهـا في الخطوة السابقة.
3) **تكرار الخطوات التالية حتى الوصول إلى أدنى مستوى ممكن للمخططـات التفصيلية:**

أ) **تقسيم العملية إلى عدة عمليات فرعية ورسم المسودة الأولى للمخطط،** مع تبيان هذه العمليات وتدفقات البيانات وينصح بالبدء من التدفقات الداخلة باتجاه العمليات وانتهاء بالتدفقات الخارجة كما يجب التقيد باتجاه المخطط من اليسار نحو اليمين.

ب) **مراجعة مسودة المخطط وتبسيطه** حتى الوصول إلى مخطط تدفق بيانات واضح وسهل الفهم، يمكن تكرار عملية المراجعة عدة مرات.

جـ) **التأكد من خلو المخطط من الأخطاء** والتي يمكن أن تكون:

- وجود تدفقات كثيرة ومتشابكة، في هذه الحالة يجب إعادة تجزئة العمليات بطريقة أخرى، بحيث تعكس حجما متقاربا من العمل في نفس المستوى.
- وجود مخرجات ليس لها مدخلات وكذلك وجود مدخلات غير مستخدمة من قبل العمليات، في هذه الحالة يجب إعادة النظر في المخطط.
- وجود تدفقات بيانات أو عمليات يصعب تحديد أسماء واصفة لها، وهذا قد يكون ناتجا عن تركيبتها غير المنطقية ولذلك يجب إعادة تقسيمها (تجزئتها) بطريقة مناسبة.
- وجود تفاوت بين العمليات في نفس المستوى من حيث عدد المستويات التفصيلية لها، فمثلا وجود في المخطط العام عملية لا يمكن تقسيمها وعملية أخرى يتطلب تفصيلها عدة مستويات أدنى، في هذه الحالة يجب إعادة النظر في تقسيم العمليات.

د) **رسم النسخة النهائية للمخطط** التي تم التوصل إليها عند المستوى الحالي.

هـ) **مراجعة النسخة النهائية** للمستوى الحالي مع المستخدمين المعنيين.

4) **ترتيب النسخ النهائية للمخططات** في جميع المستويات بشكل منطقي **والتأكد من صحة ترقيم عملياتها** ومن **الموازنة الصحيحة** لتدفقات البيانات في المستويات المختلفة.

5) **المراجعة العامة للمخططات** جميعا مع الإدارة والمستخدمين للتأكد من صحة تمثيلها للنظام الحالي، وإجراء التعديلات اللازمة عند الضرورة.

أسئلة الفصل

1- ما هي المنهجية الهيكلية لتحليل الأنظمة؟

2- عدد واشرح المكونات الأربعة لمخططات تدفق البيانات؟

3- اشرح المستويات المختلفة لمخططات تدفق البيانات؟

4- ما هو المخطط البيئي لتدفق بيانات النظام وما هي المعلومات التي يعكسها؟

5- ما هي أهم المعلومات التي يعكسها المخطط العام لتدفق بيانات النظام؟

6- عدد أهم المتطلبات التي يجب أن تتوفر في مخططات تدفق البيانات؟

7- عدد واشرح خصائص مخططات البيانات الجيدة؟

8- ما المقصود بمبدأ المحافظة على البيانات (حفظ البيانات)؟

9- كيف يتم ترقيم العمليات في مخططات التدفق وما هي أهمية ذلك؟

10- ما الفرق بين مخططات تدفق البيانات المادية والمنطقية؟ وما هي الخطوات اللازمة لتحويل المخططات المادية إلى مخططات منطقية؟

11- اشرح خطوات رسم مخططات تدفق البيانات؟

تمارين

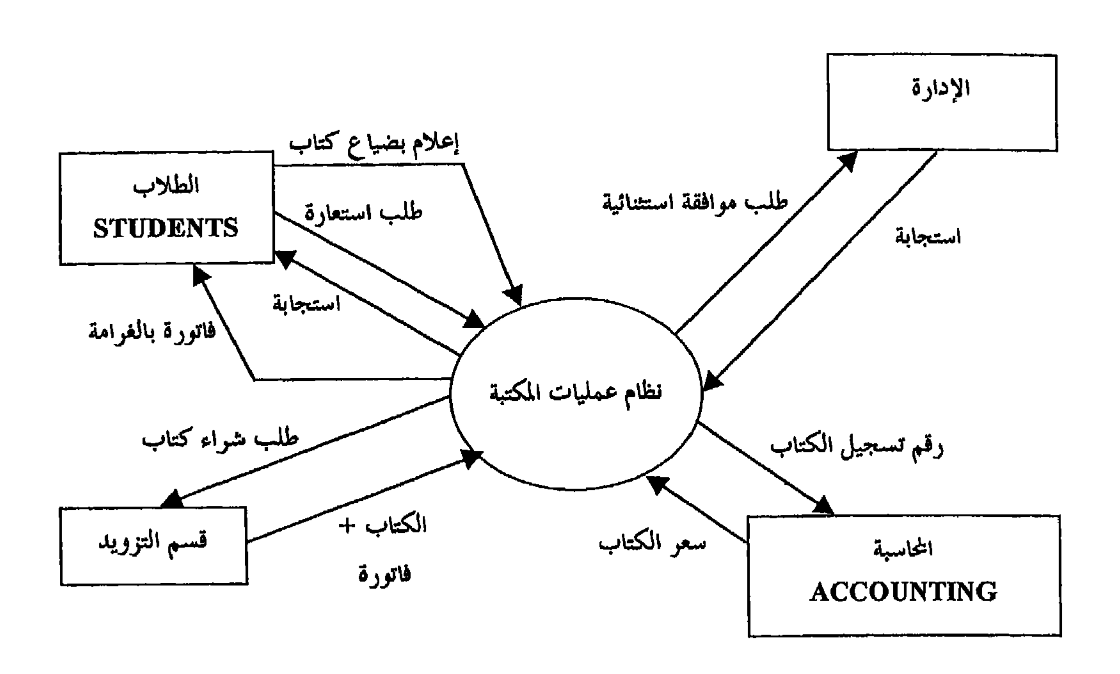
1)من خلال معرفتك لعملية التسجيل التي تقوم بها كل فصل ارسم المخططات التالية:

أ) المخطط البيئي لنظام التسجيل.

ب) المخطط العام لنظام التسجيل وحدد فيه العمليات الرئيسية والملفات المستخدمة وتدفقات البيانات فيما بينها؟

جـ) المخطط التفصيلي لتدفق البيانات لإحدى العمليات الرئيسية في المخطط العام؟

2)ارسم المخطط العام لتدفق البيانات لنظام عمليات المكتبة الجامعية في ضوء المخطط البيئي التالي:



3) ارسم مخططات تدفق البيانات (البيئي والعام) لنظام معلومات التسويق وفق المعلومات التالية:

الوظائف	المدخلات	المخرجات
عرض المنتجات	المنتجات المتاحة (من المنتجين)	سجلات في ملف المنتجات
الاستعلام عن المنتجات	• استعلام عن المنتجات (من المستهلكين) • السجلات ذات العلاقة من ملف المنتجات	احتياجات المستهلكين
الشراء الأولي	• احتياجات المستهلكين	طلبات الشراء
ترتيب عملية الشحن	طلبات الشراء	متطلبات الشحن
جدولة عمليات الشحن	متطلبات الشحن	ترتيبات الشحن
إعداد الفواتير	• المنتجات المرسلة (من المنتجين) • السجلات ذات العلاقة في ملف المنتجات	فاتورة إلى المستهلك (المشتري)

3)ارسم مخطط تدفق البيانات (البيئي والعام) لنظام الرواتب والأجور الذي يقـــوم بالوظائف التالية:

- إعداد شيكات رواتب للموظفين.
- إعداد تقرير شهري للإدارة.
- إعداد تقرير بالضرائب المستحقة.
- استلام الطلبات الموافق عليها من قبل الإدارة.
- إضافة سجلات الموظفين الجدد إلى ملف الرواتب.
- تعديل السجلات الموجودة في ملف الرواتب بناء على طلبات الإدارة.
- استلام وتدقيق وتسجيل المدخلات المتعلقة بالعمليات الخاصـــة بـــالرواتب (إجازات مرضية –حسميات – مكافئات وغيرها)
- حذف سجلات الموظفين الذين تركوا العمل من ملف الرواتب.

الفصل السابع

تحليل بيانات النظام

يهدف تحليل البيانات **Data analysis** إلى إعداد ما يسمى بنموذج بيانات النظام **Data model** . وتعتبر هذه العملية من الأنشطة الرئيسية لمرحلة التحليل . وتتم نمذجة البيانات غالباً باستخدام النماذج البيانية ، أي المخططات والرسوم التي تشبه إلى حد ما مخططات تدفق البيانات التي تعرفنا إليها في الفصل السابق .

1- **خطوات نمذجة البيانات:** وكما هو الحال عند نمذجة العمليات وإعداد مخططات تدفق البيانات فإن نمذجة البيانات تتم عادة في ثلاثة خطوات هي:

أ) إعداد النموذج المفاهيمي للبيانات **Conceptual data model** :

وتسمى هذه الخطوة أيضاً نمذجة بيانات النظام، ويتم خلالها بناء النموذج الذي يعكس الموضوعات (الأشياء) Objects الرئيسية للبيانات، وعلاقاتها مع بعضها البعض . ويسمى التحليل في هذا المستوى بتحليل المضمون أو المعنى Semantic analysis .

ب) تحليل العلاقات **Relational analysis** :

ويتم فيه تحسين النموذج المفاهيمي بإعادة تصميم الكينونات بطريقة تقلل التكرارات وتحول الكينونات إلى علاقات مبسطة يمكن التعامل معها بمرونة

وسهولة. وتسمى هذه العملية أيضاً تسوية أو تطبيع البيانات Data Normalization وبناء النموذج العلاقاتي للبيانات Relational Data Model.

جـ) تصميم قاعدة البيانات :

وتهتم بتحويل النموذج العلاقاتي Relational Model إلى توصيف قاعدة بيانات النظام Database Definition .

ونظراً لأن الخطوة الأولى في تحليل البيانات تتم في مرحلة تحليل النظام، بينما تتم الخطوتان ، الثانية والثالثة في مرحلة التصميم ، فإننا سنقوم في هذا الفصل بدراسة الأساليب المختلفة لنمذجة البيانات.

تهدف نمذجة البيانات Data modeling إلى توصيف الخصائص الجوهرية لبيانات النظام . وتستخدم طرق وأساليب متنوعة لإعداد نماذج البيانات منها :

- مخططات هيكل البيانات Data Structure Diagrams(DSD) .
- مخططات الكينونات–العلاقات Entity – Relationship (E-RD) Diagrams .
- مخططات الكينونات Entity Diagrams .
- مخططات العلاقات الثنائية Binary Relationship Diagrams .
- مخططات تاريخ حياة الكينونة Entity Life History Diagrams
- شبكات بتري Petri Net .

وسندرس في هذا الفصل الأساليب المذكورة أعلاه :

2– مخططات هيكل البيانات Data Structure Diagrams :

تستخدم هذه المخططات لتوضيح طريقة توزع بيانات النظام في مجموعات مختلفة تسمى الكينونات Entity Sets . ويتم إعداد هذه المخططات

استناداً إلى مخططات تدفق بيانات النظام التي تمت دراستها في الفصل السابق . وكما هو مبين في الشكل (1.7) تتكون هذه المخططات من العناصر الآتية :

أ)مجموعة الكينونة Entity Set :

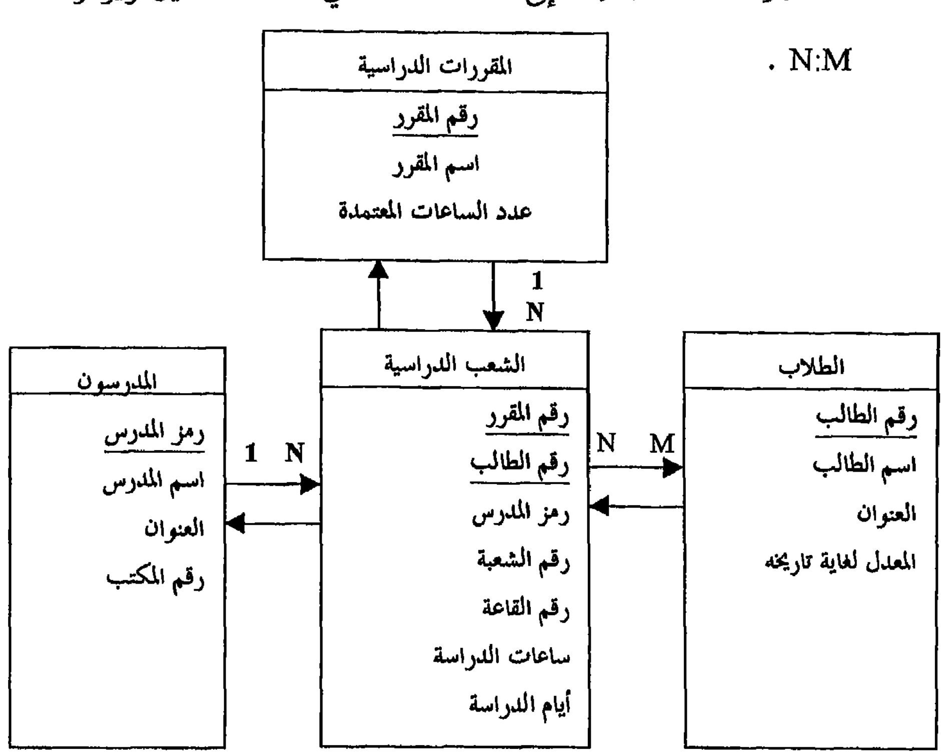
يرمز لهذه المجموعة في المخطط بشكل مستطيل يكتب في قسمه العلوي تسمية المجموعة . وتمثل الكينونة حدوث سجل واحد ضمن المجموعة ، مثلاً سجل طالب ما ضمن مجموعة كينونات (سجلات) "الطلاب" . أما حقول البيانات التي يمكن أن تتضمنها الكينونة فتسمى **خصائص أو صفات الكينونة Attributes** وتكتب ضمن المستطيل الذي يمثل الكينونة المعنية ، ويتم وضع سطر تحت الخصائص المفتاحية فيها والتي يمكن أن تكون حقلاً بسيطاً Simple Key أو مركباً Concatenated Key .

ب)الروابط أو المؤشرات المنطقية Logical Pointers :

ترتبط الكينونات في قاعدة البيانات مع بعضها البعض بعلاقات معينة، ويتم تمثيل هذه الروابط في المخطط باستخدام أسهم تمثل العلاقات المنطقية أو الروابط بين مجموعات الكينونات، كما هو مبين في الشكل (1.7) أما الرموز التي نشاهدها في المخطط عند بداية ونهاية كل سهم فهي تمثل نوع العلاقة Cardinality، التي يمكن أن تكون إحدى العلاقات الآتية :

- علاقة سجل إلى سجل (واحد إلى واحد) في كلا الاتجاهين ويرمز لها 1.1.
- علاقة سجل إلى عدة سجلات في اتجاه معين وسجل إلى سجل في الاتجاه المعاكس وتكتب 1: N.

• علاقة عدة سجلات إلى عدة سجلات في كلا الاتجاهين ويرمز لهـــا N:M .



شكل (1.7) مخطط هيكل بيانات (مثال نظام التسجيل)

1.2 خطوات إعداد مخطط هيكل البيانات :

يمكن إنشاء مخطط هيكل البيانات للنظام الذي يجري تحليله وفق الخطــوات الآتية :

1) **تحديد مجموعات الكينونات** : لقد أشرنا أعـــلاه أن إنشـــاء هـــذه المخططات يتم بالاستناد إلى مخططات تدفق البيانات (DFD) . بالرجوع إلى هذه المخططات نجد أن مجموعات الكينونات التي يمكن تمثيل بنية البيانات مـــن خلالهـــا هي:

- مخازن البيانات الموجودة في مخططات التدفق .
- العمليات التي يتم خلالها تحديث مخازن البيانات .

وهكذا فإنه في هذه الخطوة يتم تحديد مجموعات الكينونة التي سيتم من خلالها وصف هيكل بيانات النظام . وبعد تحديد هذه المجموعات يمكن رسمها بشكل مستطيلات في ورقة منفصلة وكتابة تسمياتها في الجزء العلوي لكل مستطيل.

2)تحديد خصائص أو صفات الكينونات :

تهتم هذه الخطوة بتحديد الخصائص أو الصفات التي ستتضمنها كل كينونة. ويتم ذلك بالرجوع إلى مخططات تدفق البيانات أو قاموس بيانات النظام أو بالرجوع إلى وثائق النظام . فمثلاً لتحديد الخصائص اللازمة للكينونة التي تمثل فاتورة معينة يمكن دراسة الفاتورة نفسها لنجد أنها تتضمن حقول عديدة هي : رقم الفاتورة – تاريخ الفاتورة – اسم الزبون – تفاصيل البيانات المتعلقة بالمنتجات المباعة وكمياتها وأسعارها . وأخيراً القيمة الإجمالية للفاتورة ونسبة الحسم والقيمة الصافية وغيرها . وهذه الحقول تمثل خصائص مجموعة الكينونات الخاصة بالفواتير.

وبعد تحديد جميع خصائص كل مجموعة يجب اختيار واحدة وأكثر من هذه الخصائص لتكون بمثابة خصائص مفتاحية لهذه المجموعة. ويجب أن تكون هذه الخصائص المفتاحية فريدة ضمن المجموعة، أي لا يمكن أن يكون لكينونات ضمن المجموعة الواحدة نفس قيمة الحقول المفتاحية.

وفي معظم الحالات تعتبر عملية تحديد المفاتيح سهلة وبسيطة فمثلاً مجموعة كينونات المواد يكون مفتاحها رقم المادة وهكذا. ولكن في بعض الكينونات قد نحتاج إلى استخدام اكثر من خاصية معاً لتمثيل المفتاح كما هو الحال في مجموعة كينونات الشعب في المثال المبين في الشكل (1.7).

وفي نهاية هذه الخطوة يجب سرد أسماء الخصائص المتعلقة بكل مجموعة ضمن المستطيل الذي يمثل هذه المجموعة ثم وضع خط تحت الخصائص المفتاحية، التي تكتب عادة في أعلى القائمة.

أما الخصائص المتكررة القيمة Multivalued أي تلك التي يمكن أن يكون لها أكثر من قيمة في السجل الواحد (الكينونة الواحدة)، كالبيانات التفصيلية للفاتورة التي يمكن أن تتضمن عدة أسطر تتكرر فيها قيم وأسماء وكميات وأسعار المواد المباعة، فيجب كتابتها مرة واحدة في القائمة، ويشار إلى كونها يمكن أن تتكرر أكثر من مرة بوصفها ضمن أقواس { }

3)إنشاء الروابط المنطقية:

في هذه الخطوة يتم تحديد الروابط المنطقية التي تظهر العلاقات بين مجموعات الكينونات المختلفة في النظام، ولقد ذكرنا سابقاً أن هذه الروابط ترسم في المخطط بشكل أسهم، أي تدل على اتجاه الوصول للبيانات الموجودة في القاعدة. وبعد رسم هذه الأسهم يجب كتابة رموز العلاقات التي تمثلها والتي تعرفنا إليها أعلاه وهي رمز سجل إلى سجل أو سجل إلى عدة سجلات أو عدة سجلات إلى عدة سجلات.

4)تطبيع البيانات (.تسوية البيانات)

بانتهاء الخطوة السابقة يتم الوصول إلى مخطط هيكل البيانات للنظام الذي تجري دراسته، ويسمى مثل هذا الهيكل بالشكل غير الطبيعي Unormal Form. ولذلك يجب تحويله إلى شكل طبيعي من خلال عملية تسمى تطبيع البيانات Data Normalization وهذا ما سندرسه في فصل لاحق من هذا الكتاب، أما الآن

فنكتفي بالإشارة إلى أن هذه العملية تهدف إلى تحسين هيكل البيانات من خلال تقليل التكرارات وتبسيط المجموعات الممثلة للكينونات المختلفة في النظام.

5)تحديث مخططات التدفق وقاموس البيانات:

بعد الوصول، من خلال عملية التطبيع، إلى البنية المثلى لبيانات النظام، يجب مراجعة مخططات التدفق وقاموس البيانات لتعكس التغيرات التي تمت في مخطط هيكل بيانات النظام، وبشكل عام فان ذلك يتطلب إضافة مخازن بيانات جديدة وكذلك تعديل وصف مخازن البيانات الموجودة.

3. مخططات الكينونة – العلاقة

1.3 عناصر مخططات الكينونة – العلاقة (E-RD)

يتم تمثيل بيانات النظام في هذه المخططات بشكل مجموعات كينونة ومجموعات علاقة تربط بين مجموعات الكينونة، وتستخدم في إعداد هذه المخططات ثلاثة مفاهيم أو رموز تجريدية Abstractions هي:

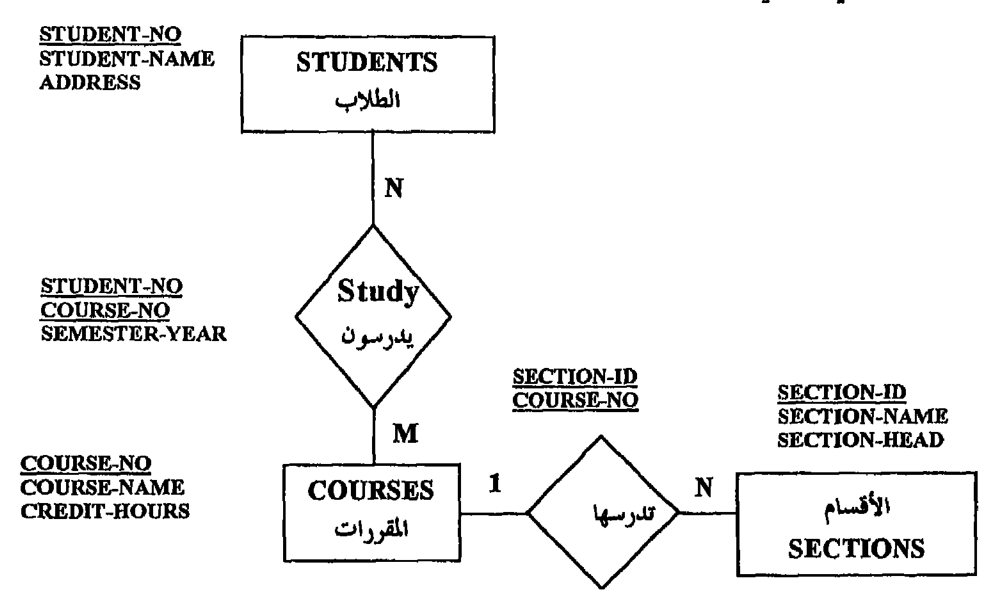
- **الكينونة Entity**: نستخدم هذا المفهوم لتمثيل أشياء محددة في النظام مثل فاتورة أو ميزانية أو أمر شراء، أو زبون أو مورد أو مادة (منتج) أو غيرها. وتسمى مجموعة الكينونات المتماثلة **مجموعة كينونة** Entity Set. فالطالب هو كينونة ، والطلاب يمكن تمثيلهم بشكل مجموعة كينونة . ويرمز لمجموعة الكينونة في المخطط بمستطيل يكتب فيه اسمها على النحو المبين في الشكل (2.7) .

ب)- العلاقة Relationship :

وتمثل تفاعل Interaction ذو معنى بين كينونتين أو أكثر . وتسمى مجموعة العلاقات المتماثلة بمجموعة علاقة Relationship Set ويرمز لها في المخطط بشكل معين يكتب داخلة اسم العلاقة الذي يجب أن يعبر عن فعل أو عمل .

جـ) الخاصية أو السمة Attribute:

وتمثل صفة أو خاصية معينة لمجموعة كينونة أو لمجموعة علاقة. وتكتب الخصائص في مخططات الكينونة – العلاقة أمام رموز المجموعات (كينونات أو علاقات) التي تنتمي إليها هذه الخصائص كما هو مبين في شكل (2.7).



شكل (2.7) مثال لمخطط كينونة–علاقة

د) **عدد العلاقات Cardinality**: وهي عدد العلاقات التي يمكن أن تظهر فيها الكينونة الواحدة، وتسمى أيضاً Relationship cardinality . حيث يمكن أن تظهر الكينونة في علاقة واحدة أو عدة علاقات.

ففي الشكل 2.7 أعلاه يمكن أن يظهر رقم الطالب الواحد (الكينونة الواحدة) في عدة علاقات حسب عدد المقررات التي سجل فيها. ولذلك يكتب الحرف N أمام الخط الذي يصل بين مجموعة الطالب ومجموعة العلاقة "يدرس" وكذلك الأمر من جهة مجموعة الكينونة " المقررات" يمكن أن يظهر رقم مقرر معين في عدد من العلاقات بحسب عدد الطلاب الذين سجلوا في هذا المقرر، ولذلك نكتب (M) أمام الخط الذي يصل بين هاتين المجموعتين. أما عدد العلاقات في مجموعة " تدرسها" التي يمكن أن تظهر لمقرر واحد فهي علاقة واحدة فقط، لان المقرر يتم تصميمه وتدريسه من قبل قسم واحد فقط يكون مسؤولا عن ذلك. ولذلك نجد الرقم 1 أمام الخط الواصل بين المجموعة "المقررات" ومجموعة "تدرسها".

هـ) مشاركة الكينونة في مجموعة العلاقة:

يمكن أن تشارك الكينونة الواحدة في مجموعة العلاقة بعدة طرق :

- **مشاركة إلزامية Mandatory Participation** : عندما يكون من الضروري أن تظهر الكينونة على الأقل مرة واحدة في مجموعة العلاقة . فمثلاً لكي يتم تسجيل الطالب في مجموعة الطلاب يجب أن يتم تسجيله أيضاً في مقرر واحد على الأقل .

مشاركة اختيارية : Optional عندما يكون ليس من الضروري أن يسجل الطالب في أي مقرر ليكون اسمه في مجموعة الطلاب يرمز للعلاقة الاختيارية بدائرة صغيرة كما هو مبين في الشكل(2.7).

2.3 قواعد رسم مخططات الكينونة – العلاقة.

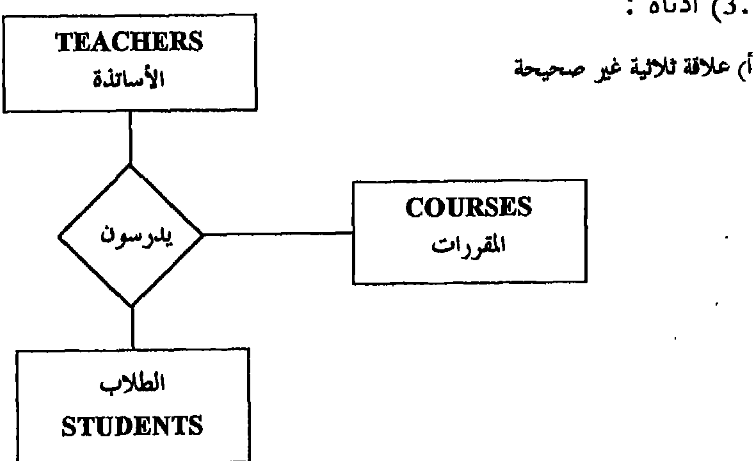
لقد تعرفنا حتى الآن إلى العناصر الأساسية التي تتكون منها بنية مخططات الكينونة – العلاقة، وهي مجموعات الكينونة ومجموعات العلاقة والخصائص. وعدد العلاقات ونوع المشاركة.

وسندرس فيما يلي القواعد التي يتم في ضوءها إعداد هذه المخططات

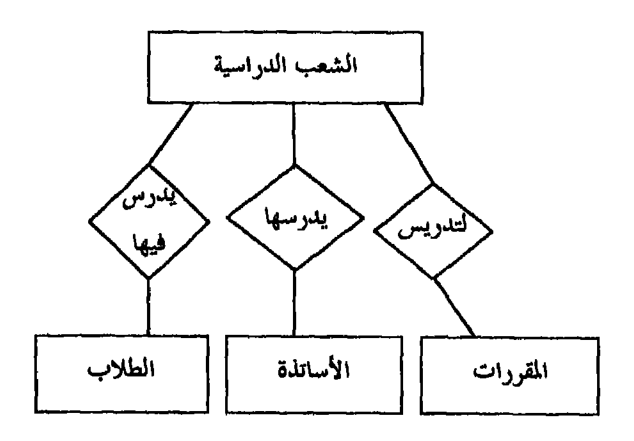
- تمثل مخططات الكينونة – العلاقة مجموعات الكينونة ومجموعات العلاقات الموجودة في النظام .
- تستخدم مجموعة العلاقات لربط مجموعتي كينونات كما هو مبين في الشكل (1.7) أعلاه . حيث استخدمت مجموعة العلاقة " يدرسون " لربط مجموعة الكينونة "الطلاب" مع مجموعة الكينونة " المقررات" . ولذلك تسمى هذه المجموعة أيضاً Binary Relationship أي مجموعة ثنائية .

في حال وجود مجموعة علاقة تربط بين أكثر من مجموعتين كينونة فإن مثل هذه العلاقة غير صحيحة ويجب استبدالها بمجموعة كينونة كما هو مبين في الشكل (3.7) أدناه :

أ) علاقة ثلاثية غير صحيحة

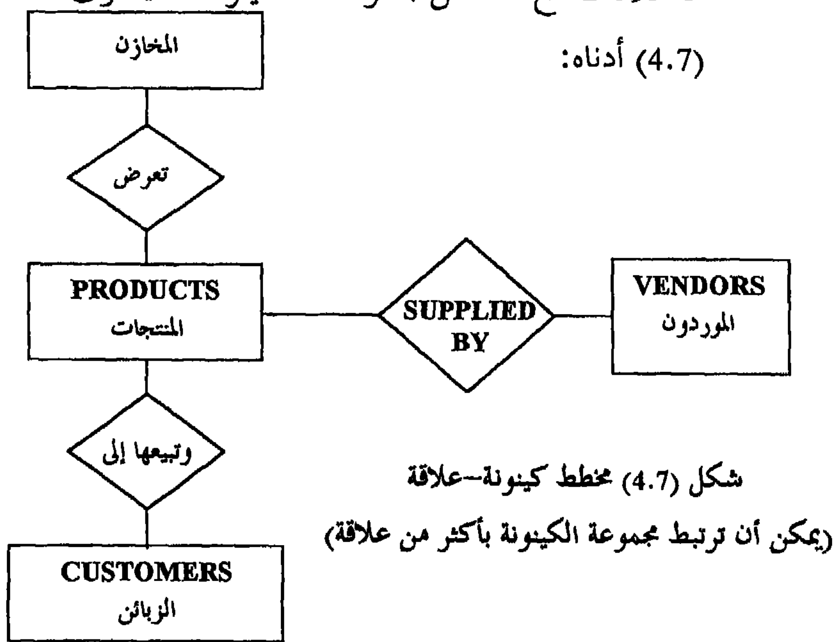


ب) تصحيح العلاقة بإنقاصها لتصبح ثنائية



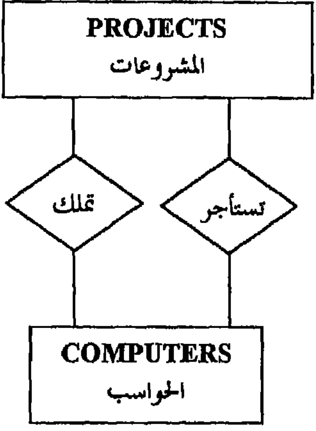
(شكل 3.7) تصحيح العلاقات الأكثر من ثنائية

• يمكن لمجموعة الكينونة أن ترتبط بعدة مجموعات علاقة ، أي أن يكون لها علاقات مع عدد من مجموعات الكينونات الأخرى . كما في الشكل (4.7) أدناه:



شكل (4.7) مخطط كينونة-علاقة
(يمكن أن ترتبط مجموعة الكينونة بأكثر من علاقة)

• يمكن أن ترتبط مجموعتين كينونة فيما بينها بأكثر من علاقة على النحو المبين في الشكل (5.7):



شكل (5.7) ربط مجموعتي كينونة بأكثر من علاقة

ومن الواضح أن السبب في ذلك هو وجود خصائص مختلفة في مجموعتي العلاقة

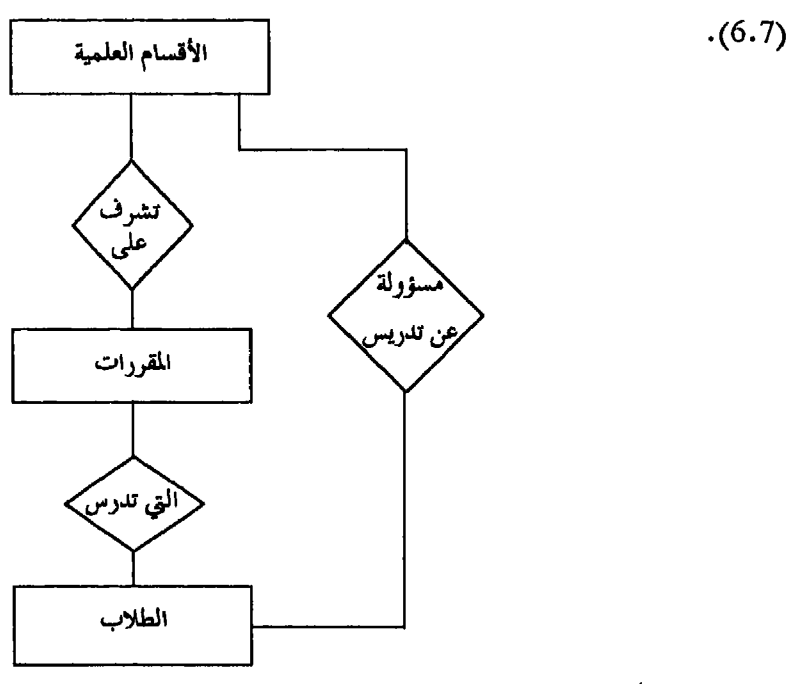
• يمكن كتابة الخصائص في مخططات الكينونة – العلاقة بطرق مختلفة فمثلا :

- بجانب المجموعات التي تنتمي إليها هذه الخصائص (المستطيلات أو المعينات التي تمثل هذه المجموعات) .
- ضمن دائرة متصلة بخط إلى المستطيل أو المعين الذي يمثل المجموعة التي تنتمي إليها هذه الخصائص .
- سردها بشكل قوائم منفصلة لكل مجموعة في اسفل المخطط .

• يجب اختيار أحد خصائص مجموعة الكينونة ليكون المعرف Identifier،والذي يجب أن تكون له قيمة فريدة غير متكررة في المجموعة Unique Value. ويوضع تحته خط للدلالة عليه ، كما هو مبين في الشكل

(2.7) أما معرف مجموعة العلاقة فيكون عادة حقلاً مركباً يتكون من معرفي مجموعتي الكينونة التي تربط بينهما مجموعة العلاقة هذه (انظر الشكل (2.7)) . وكذلك فإن معرف مجموعة الكينونة التابعة يجب أن يكون مركباً .

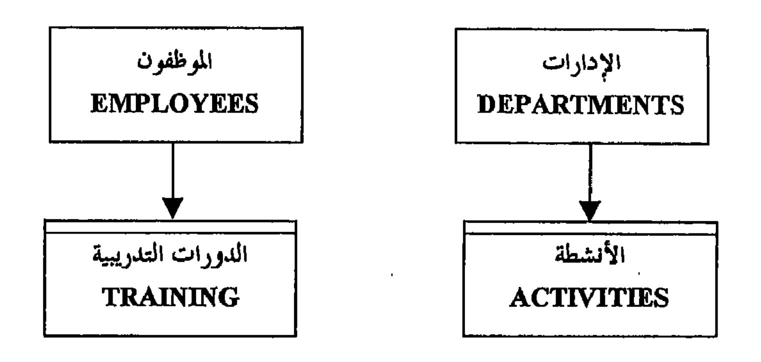
- يفضل أن لا تحتوي مخططات الكينونة العلاقة على مجموعة علاقة يمكن اشتقاقها من مجموعات علاقة أخرى كما هو مبين في الشكل (6.7).



شكل (6.7) يجب تجنب العلاقات التي يمكن اشتقاقها في مخططات الكينونة– العلاقة

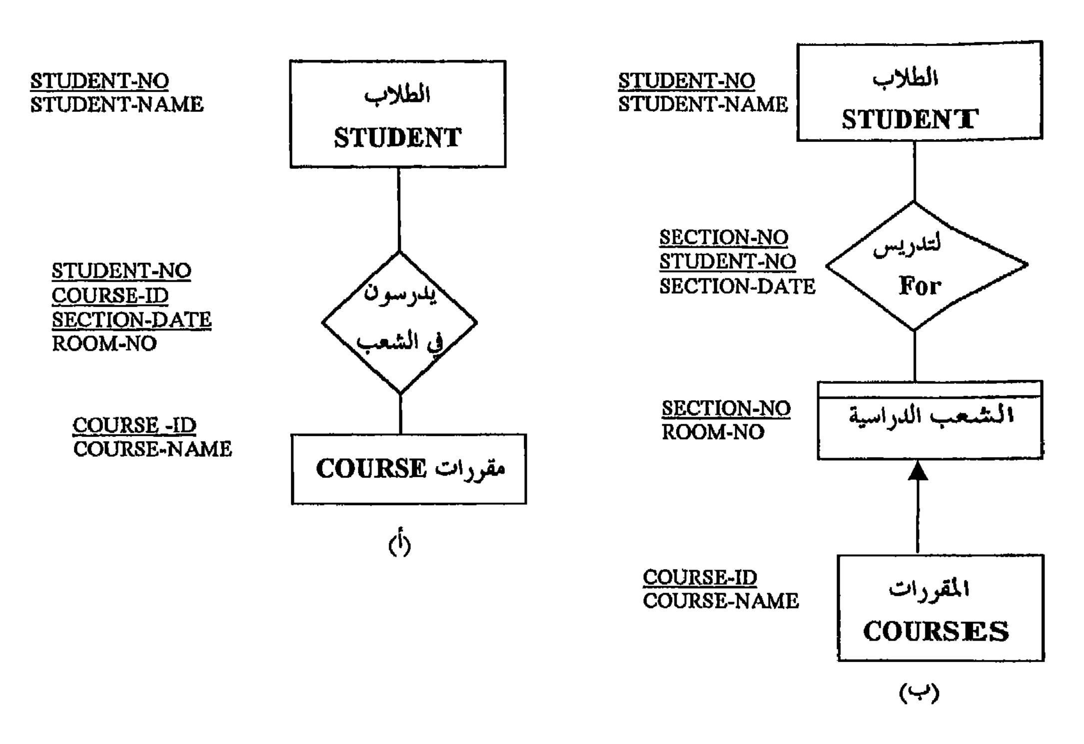
3.3 الأشكال المختلفة لمجموعات الكينونة Entity Sets Variations :

تمثل الكينونة من حيث المبدأ ، شيئا أو مفهوما مستقلا أي أن وجودها لا يرتبط بوجود أي شيء أو عنصرا آخر في النظام . فوجود طالب ما في كينونة الطلاب لا يرتبط بأي شيء . ولكن في بعض الحالات قد يكون وجود الكينونة في مجموعة ما مرتبط بوجود كينونة أخرى في مجموعة ثانية . وتسمى مثل هذه الكينونة **بالكينونة التابعة** Dependent Entry فمثلا في الشكل (7.7) لا يمكن أن تتواجد الكينونة الخاصة بإجازة موظف ما في كينونة "الدورات التدريبية" إذا لم تكن هناك كينونة خاصة لهذا الموظف في مجموعة كينونة " الموظفون " . وكذلك الحال بالنسبة لمجموعة كينونة " الأنشطة " فهي تابعة لمجموعة كينونة " الإدارات " . ويرمز لمجموعة الكينونة التابعة بمستطيل له خطان في أعلاه وبسهم ينطلق من المجموعة الرئيسية إلى المجموعة التابعة. كما هو مبين في الشكل (7.7) أدناه:



الشكل(7.7) مجموعة الكينونة المستقلة ومجموعة الكينونة التابعة

يستخدم مفهوم الكينونة التابعة لتمثيل البيانات التفصيلية أو المتكـــررة أو التاريخية المرتبطة بعناصر النظام. ويقصد بالبيانات التاريخية تلك البيانات المتكـــررة والتي تربط بين مجموعتي كينونة، فمثلاً بيانات الجدول الزمني لدروس الشعبة هـــي عبارة عن بيانات تربط بين مجموعة كينونة "الطلاب" ومجموعة كينونة المقـــررات، كما هو مبين في الشكل (8.7) أدناه:

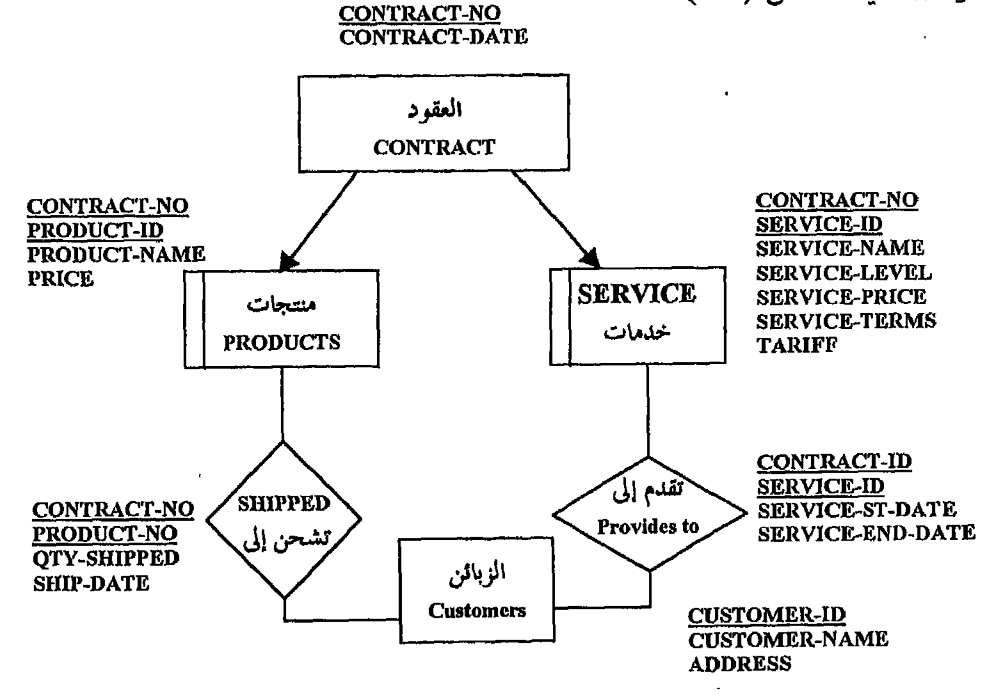


الشكل(8.7) مفهوم العلاقات المتعددة واستخدام مجموعة الكينونات التابعة لمعالجتها

نلاحظ في الجزء (أ) من المخطط (8.7) أعلاه أنه يمكن أن يرتبـــط نفـــس الطالـب ونفس المقرر (الكينونات) بعدة علاقات حسب عدد الجلسات التدريســـية التي يتضمنها المقرر، وتسمى مثل هذه العلاقة التي يمكن أن تتكـــرر بـــين نفـــس

الكينونتين العلاقة المتعددة Multiple Relationship ، ونظراً لأن مثل هذه العلاقات تتضمن تكراراً كبيراً للبيانات فانه يفضل معالجتها باستخدام الكينونة التابعة والمسماة " الشعب الدراسية" ثم ربط هذه الكينونة التابعة مع كينونات المقررات.

كما تجدر الإشارة إلي أن جميع الكينونات الموجودة ضمن مجموعة واحدة يجب أن يتم معالجتها بنفس الطريقة، أي أن عمليات المعالجة الخاصة بمجموعة كينونة يجب أن يتم تطبيقها على جميع الكينونات الموجودة في هذه المجموعة، وفي حال وجود مجموعة كينونة تتطلب كينونتها معالجات مختلفة فانه يجب تقسيمها إلى **مجموعات فرعية Subsets**. ترتبط مع المجموعة الرئيسة بنفس طريقة ربط مجموعات الكينونة التابعة، وترسم بشكل مستطيل فيه خطان عموديين في جانبه الأيسر، كما هو مبين في الشكل (9.7).

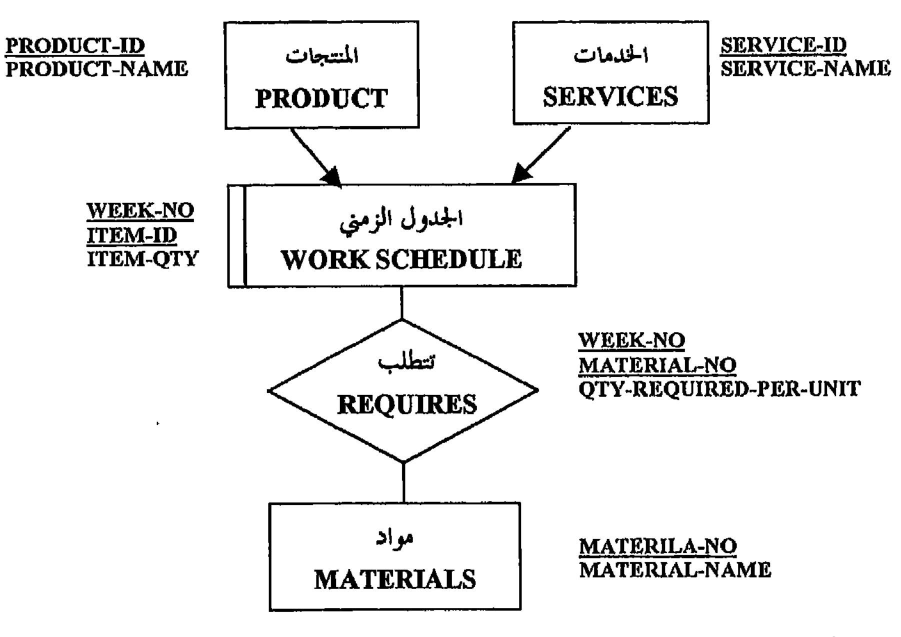


شكل (9.7) مخطط E-RD يضم مجموعات فرعية

يتبين من الشكل (9.7) أعلاه أنه يتم توزيع خصائص مجموعة الكينونة بين المجموعة الأم Parent Set وبين المجموعة الفرعية Subset، فالخصائص المشتركة بين جميع الكينونات توضع في المجموعة الأم، أما الخصائص الغير مشتركة فيتم وضعها في مجموعات فرعية حسب الموضوعات التي يمكن أن تنتمي إليها.

لذلك تم تحديد رقم العقد وتاريخه كخصائص (صفات) للمجموعة الأم، ونظراً لأن المبيعات يمكن أن تتعلق بخدمات أو منتجات وكل منها يتطلب معالجة خاصة، فقد قمنا بتحديد مجموعة فرعية للمنتجات وأخرى للخدمات، وأعطيت هاتين المجموعتين الخصائص (الصفات) التي تميزهما. ونلاحظ كذلك من الشكل (9.7) استخدام معرف المجموعة الأم كمعرف للمجموعات الفرعية التابعة لها.

كما يمكن أن تتبع المجموعة الفرعية الواحدة إلى اكثر من مجموعة رئيسه كما هو مبين في الشكل (10.7) أدناه:



شكل (7.10) المجموعة الفرعية يمكن أن تتبع لعدة مجموعات رئيسية

4.3 خطوات إعداد مخططات الكينونة – العلاقة:

يقوم محلل الأنظمة بإعداد هذه المخططات خلال عملية التحليل ويتم إعداد هذه المخططات وفقاً للخطوات التالية:

1) تحديد الكينونات Define Entities:

في هذه الخطوة يتم تحديد عناصر البيئة والموارد، والعمليات Transactions التي سيتم وصفها بواسطة بيانات النظام، ويتم نمذجة كل عنصر من هـــذه العناصر بشكل مجموعة كينونة واحدة، فمثلاً يمكن تمثيل الطلاب بشـــكل مجموعة كينونة نسميها " الطلاب" ومجموعة المقررات في كينونة أخـــرى نسميها المقررات وهكذا، والقاعدة هنا أنه لا يمكـــن جمـــع مفـــاهيم أو عناصـــر متعددة بمجموعة كينونة واحدة، لأن المفاهيم المتعددة سيكون لها خصائص مختلفة، ولذلك يجب أن تمثل الكينونة مفهوماً (عنصراً) واحـــداً فقط، ويتم تسمية الكينونات بأسماء الأشياء التي تمثلـــها كالمنتجـــات أو الزبائن أو الموظفين أو الطلاب.

وبشكل عام فان المفاهيم أو العناصر الموجودة في النظام الذي يجري تحليله، والتي يمكن أن تكون مجموعة كينونة هي:

- الأشياء أو العناصر المادية: كالمنتجات والطلاب والأفـــراد والزبـــائن وغيرها.
- العناصر أو البنى التنظيميـــة : كالمشـــروعات والأقســـام والإدارات والموازنات والمخازن وغيرها.
- العمليات التي يمكن أن تحـــدث في النظـــام: كعمليـــات الإنتـــاج أو الصيانـــة، أو الاستلام أو التوظيف أو غيرها.

2) تحديد العلاقات Define the Relationships

في هذه الخطوة يتم تحديد العلاقات التي يمكن أن تربط مجموعات الكينونة، التي تم تحديدها في الخطوة السابقة، مع بعضها البعض. وتمثل هذه العلاقة غالبا نشاط أو عمل ما، فالطلاب يدرسون المقررات، والزبائن يشترون المنتجات ولذلك نحدد مجموعة علاقة باسم " يدرسون" لربط مجموعة كينونة "الطلاب" مع مجموعة كينونة "المقررات". وهكذا يجري في هذه المرحلة البحث عن جميع العلاقات اللازمة لتفاعل مجموعات الكينونة الموجودة في النظام مع بعضها البعض. ويتم تسمية مجموعات العلاقات باستخدام الأفعال التي تعبر عن عملية التفاعل بين الكينونات، وأحيانا يمكن استخدام حروف الجر لتسمية هذه العلاقات.

3) رسم مسودة مخطط الكينونة – العلاقة

بعد تحديد مجموعات الكينونة ومجموعات العلاقة يجب إعداد رسم أولي أو مسودة لمخطط الكينونة – العلاقة يتم فيه تمثيل مجموعات الكينونة بمستطيلات ومجموعات العلاقة بمعينات ورسم الخطوط التي تربط فيما بينها. هنا يجب الحرص على المحافظة قدر الإمكان على الاتجاهات المتعارف عليها بشكل عام في جميع أنواع المخططات، وهي الاتجاه من اليسار نحو اليمين ومن الأعلى نحو الأسفل وذلك بالرغم من إمكانية قراءة العلاقات في كلا الاتجاهين.

4) تحديد الخصائص Define Attributes

بعد الانتهاء من رسم مسودة المخطط يجب تحديد الخصائص التي يجب استخدامها لوصف مجموعات الكينونة ومجموعات العلاقة، وهذا يعني

اختيار عناصر البيانات Data Elements التي سيتم استخدامها لوصف هـذه المجموعات، ويتم كتابة تسميات هذه الخصـائص (عنـاصر البيانـات) بجانـب رمز كل مجموعة كينونة أو علاقة.

وعند اختيار هذه الخصائص يجب أن نحرص على أن تعبر كل خاصية عـن مفهوم بسيط، فمثلاً نستخدم لوصف مجموعـة كينونـة " المنتجـات" الخصائص التالية: رمز المنتج – اسم المنتج – وحدة القياس، وهكذا. أمـا الخصائص التي يمكن أن يكون لها قيم متعـددة Multivalued Attribute وكذلك الخصائص المركبة Structured Value فيتم كتابتها في هذا المخطط الأولي آخذين بعين الاعتبار ضرورة استبدالها في المخطط النهائي بعد إجراء عملية تطبيع البيانات Data Normalization. ففي مثالنا السـابق حـول مجموعة كينونة " المنتجات" تكون الخاصية المسماة " اسم المورد" ذات قيـم متعددة لأن المنتج الواحد يمكن أن يكون له عدة موردين، لذلـك يجـب استبدالها في المخطط النهائي، وكذلك الأمر بالنسبة للخصـائص المركبـة Structured كالعنوان مثلاً الذي يمكن تقسيمه إلى عدة حقول مثل العنوان البريدي وأرقام الهواتف والفاكس وعنوان البريد الإلكتروني وهكذا. كمـا يمكن تقسيم العنوان البريدي إلى حقول مثل اسم المدينة والرمز الـبريدي واسم الشارع وغيرها، هذه الخصائص المركبة يجب استبدالها عند عمليـة التطبيع في الخطوة التالية، وعند الانتهاء من تحديد الخصائص وكتابتها يتـم اختيار المعرفات Identifier اللازمة لكل مجموعة ووضع خط تحت أسمائـها في المخطط كما يجب تحديد نـوع الروابـط Cardinality وكتابتـها في أماكنها في المخطط.

Converted by Tiff Combine - (no stamps are applied by registered version)

5)تحسين النموذج Refining the model

بعد الانتهاء من إعداد مسودة الكينونة – العلاقة من خلال الخطوات الأربعة المذكورة أعلاه، يجب القيام بتحسين هذا النموذج الأولي، وتسمى هذه العملية تطبيع البيانات Data Normalization، وهي تهدف إلى إيجاد البنية الأكثر كفاءة لقاعدة بيانات النظام. وتتم عملية التطبيع هذه على مراحل يتم في كل مرحلة منها الوصول إلى مستوى تطبيع افضل من المستوى السابق، وتسمى هذه المستويات بالشكل الطبيعي الأول والشكل الطبيعي الثاني فالثالث وهكذا.

وسندرس عملية التطبيع هذه عند دراسة مرحلة التصميم لكونها تعتبر من أنشطة التصميم.

6)إعداد المخطط النهائي Preparing the Final E-RD

في ضوء نتائج عملية التطبيع التي يتم التوصل إليها يجب إعادة رسم مخطط الكينونة – العلاقة ليعكس البنية الجديدة لقاعدة البيانات وبذلك يتم التوصل إلى المخطط النهائي أو النموذج النهائي لبيانات النظام.

7)مراجعة وتدقيق مخطط الكينونة – العلاقة:

في هذه المرحلة الأخيرة يقوم المحلل بمراجعة المخطط بالاشتراك مع الإدارة والمستخدمين الرئيسين، وذلك بهدف التأكد من صحة ودقة تمثيله لعناصر البيانات وإدخال أي تحسينات أو تعديلات ضرورية للوصول إلى بنية اكثر كفاءه لقاعدة البيانات، التي سيتم تطويرها استناداً إلى هذا المخطط.

5.3 مخططات تدفق البيانات (DFD) و مخططات الكينونة – العلاقة (E-RD):

من الواضح أن كل من مخططات تدفق البيانات ومخططات الكينونة – العلاقة يقومان بنفس الوظيفة وهي نمذجة النظام الذي تجري دراسته ولكن من وجهات نظر مختلفة، فمخطط تدفق البيانات يقوم بنمذجة النظام من خلال التركيز على عمليات النظام Process Modeling وتحديد مدخلاتها ومخرجاتها، بينما تركز مخططات الكينونة– العلاقة على بنية بيانات النظام، ولذلك تستخدم مخططات تدفق البيانات كأساس لتحديد العمليات التي سيتم حوسبتها، وبالتالي توصيف برامج الحاسوب اللازمة لذلك، أما العمليات التي ستبقى يدوية فيتم إعداد أدلة إجراءات تشرح للمستخدم كيفية تنفيذها. أما مخطط الكينونة–العلاقة فيستخدم بشكل رئيسي لتوصيف هيكل البيانات الذي سيتم في ضوئه تصميم قاعدة بيانات النظام.

وهكذا نجد أن كل مخطط يكمل المخطط الآخر في وصف النظام الذي تجري دراسته، ويساعد في تدقيق وتحسين المخطط الآخر، وفي معظم منهجيات التحليل يتم البدء بإعداد مخططات تدفق البيانات للنظام ثم تستخدم هذه المخططات لإعداد مخطط الكينونة – العلاقة، ولكن المنهجيات الحديثة تنصح بإعداد هذين النوعين من المخططات بشكل متوازي، أي أن نمذجة العمليات ونمذجة البيانات يجب أن تتم في نفس الوقت ويبين الشكل (11.7) عناصر مخطط تدفق البيانات وما يمكن أن يكافئها من عناصر مخطط الكينونة –العلاقة.

عناصر مخططات تدفق البيانات	عناصر مخطط الكينونة – العلاقة
الكينونة الخارجية External Entity	مجموعة كينونة Entity Set
مخزن البيانات Data Store	مجموعة كينونة Entity Set
العملية Process	مجموعة علاقة Relation ship Set
تدفق البيانات Data Flow	خاصية أو صفة Attributes

جدول (11.7) عناصر مخططات **DFD** وما يمكن أن يكافئها من عناصر **ERD**.

وهكذا فكلا المخططان ضروري ويمثلان بالإضافة إلى توصيف العمليات وقاموس البيانات ما نسميه بنموذج النظام System Model. ونظرا للارتباط المتبادل بين هذه العناصر الأربعة (مخططات DFD ومخططات E-RD. وتوصيف العمليات وقاموس البيانات فان التسميات المستخدمة فيها يجب أن تكون موحدة.

4) مخططات الكينونات Entity Models

بالإضافة إلى المخططات التي تعرفنا أليها أعلاه فانه تستخدم أيضا أساليب وأدوات متعددة لنمذجة بيانات النظام، ولقد ذكرنا سابقا أن مخططات الكينونة – العلاقة (E-RD) هي الأكثر انتشارا في النمذجة، ولكن يجب على المحلل أن يعرف هذه الطرق المختلفة ليستطيع التعامل معها في جميع الحالات التي يمكن أن تواجهه.

تعتبر مخططات الكينونة Entity Models أكثر بساطة من المخططات E-RD لكونها لا تميز بين مجموعات الكينونات ومجموعات العلاقة، فكلاهما يعتبر

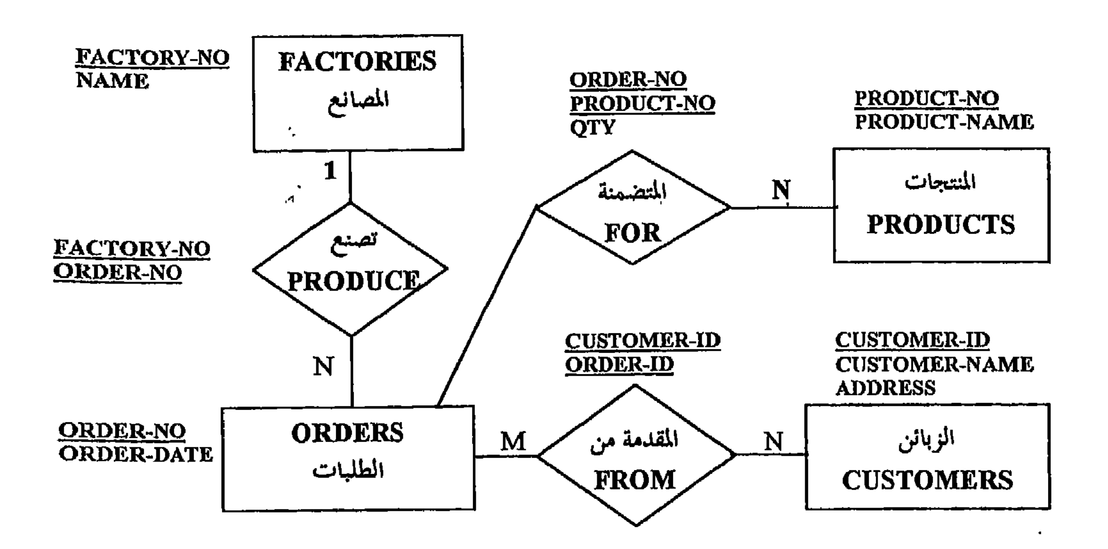
مجموعة كينونات ويتم رسمها على هذا الأســـاس في المخطـــط بشـــكل مستطيلات كما هو مبين في الشكل (12.7) وبمقارنة هذا المخطـــط مـــع مخطط الكينونة – العلاقة الموجودة فوقه نجد أن:

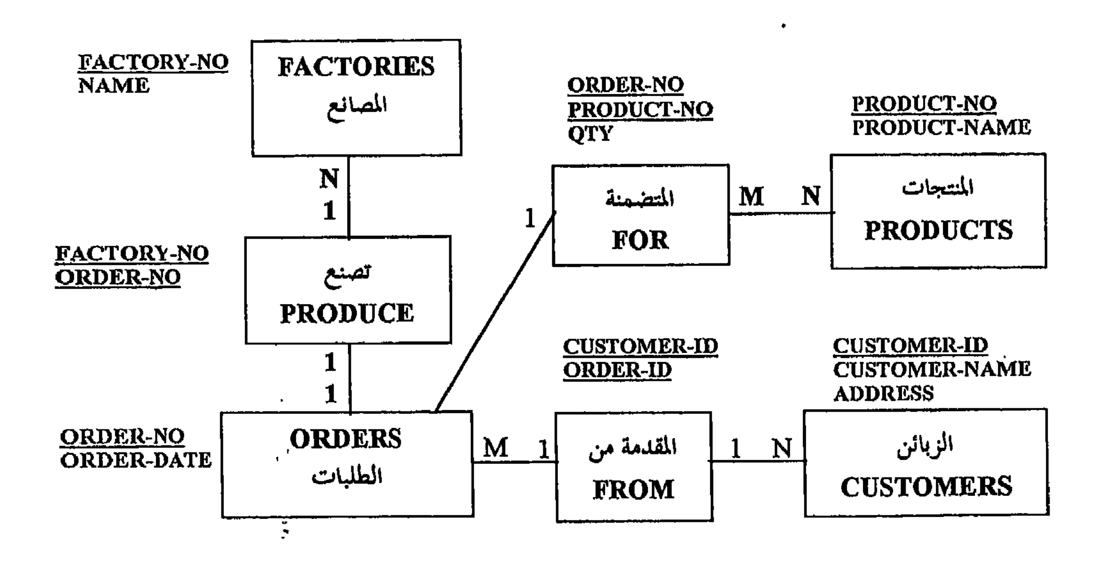
- يتم ربط الكينونات مع بعضها البعـــض بخطـــوط تســـمى في هـــذه المخططات بالروابط Associations.
- يتم تمثيل عدد العلاقات Cardinality بشكل مختلف عن مخطط الكينونة – العلاقة (E-RD) حيث يبـــين مفـــهوم الـــ Cardinality في هـــذه المخططات عدد المرات التي يمكن أن ترتبط فيـــها كينونـــة موجـــودة في مجموعة كينونة ما بكينونات موجودة في مجموعة أخرى.

فالمصنع الواحد يمكن أن يصنع العديد من الطلبات ولذلك نجد الحرف (N) فوق كينونة "المصنع"، ولكن الطلبية الواحدة لا يتم تصنيعها إلا في مصنع واحـــد فقط ولذلك نجد الرمز (1) فوق الكينونة " تصنع" وهكذا.

5) مخططات العلاقة الثنائية Binary Relationship Diagrams

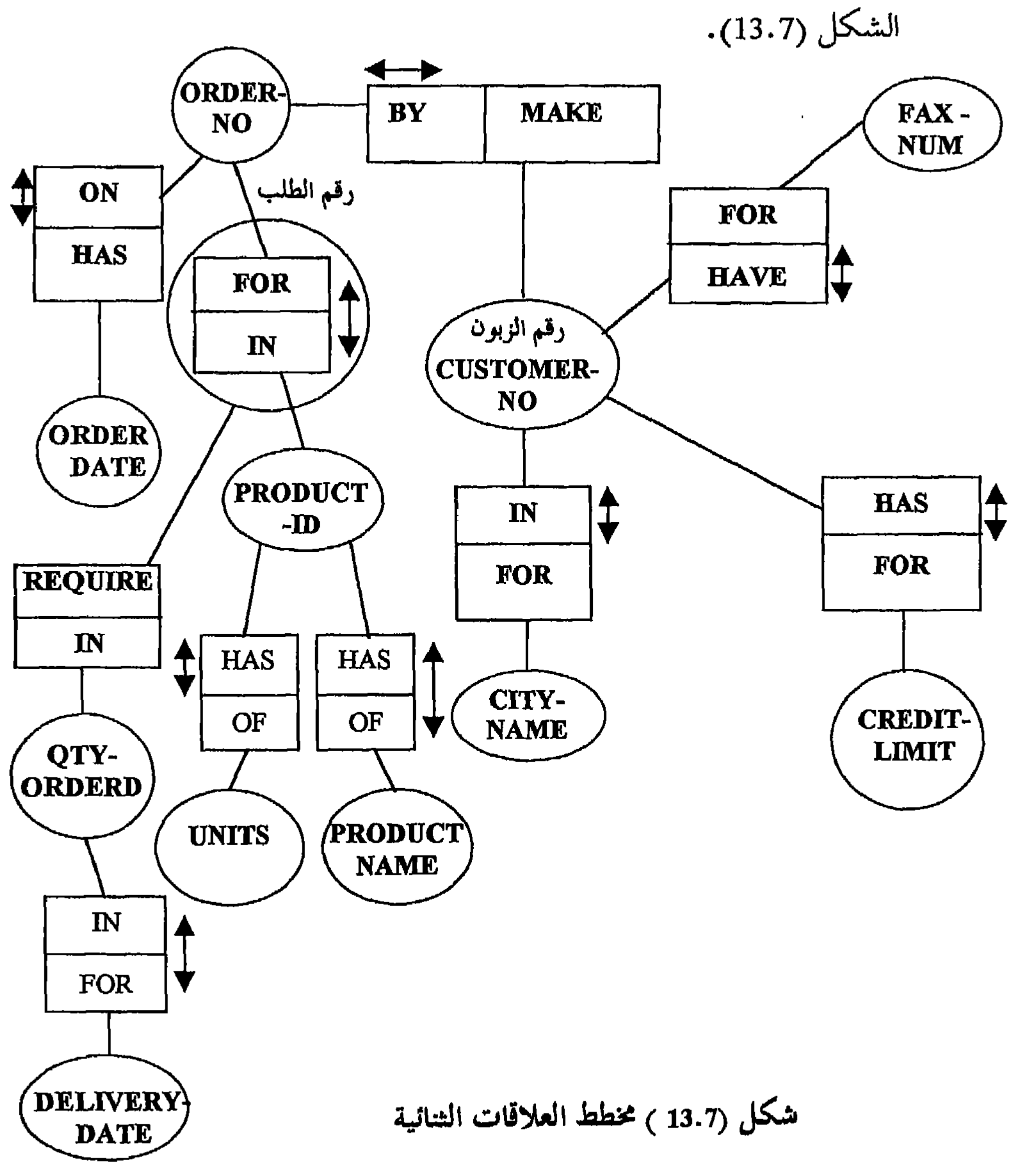
يستخدم هذا الأسلوب لنمذجة البيانات في الحالات التي تتطلب تحليـــلات معمقة ليس فقط على مستوى الكينونات وعلاقاتها أو روابطـــها، وإنمـــا علـــى مستوى الخصائص (الصفات) Attributes. فكل خاصية يحتـــاج إليها النظام هي عبارة عن مجموعه يتم تمثيلها في المخطط بشـــكل دائـــرة وتسمى بمجموعة Set أو بمجموعة قيم Values Set. ويتم ربط القيم الموجودة في مجمـــوعة معينة مع قيمة أو اكثر موجودة في المجموعات الأخرى، ويتم





شكل (12.7) مقارنة مخطط الكينونة – العلاقة ومخطط الكينونة .

رسم هذه العلاقات Associations بين القيم بخطوط تصل بين الدوائر التي تمثل مجموعات هذه القيم، وتعطى كل رابطه أو علاقة اسم يكتب في مستطيل يرسم فوق الخط الذي يمثل العلاقة ويكون مقسماً إلى جزأين يحتوي كل جزء على اسم العلاقة في أحد الاتجاهين كما هو مبين في الشكل (13.7).



شكل (13.7) مخطط العلاقات الثنائية

نجد من دراسة المخطط المشار إليه أعلاه مثلاً أن المجموعة " رقم الطلب" ترتبط بالمجموعة المسماة " رقم الزبون" كما نلاحظ تسميات هذه العلاقة في الاتجاهين حيث تقرأ على النحو التالي " رقم الطلب" الوارد من الزبون، وبالاتجاه المعاكس تقرأ " رقم الزبون الذي قدم الطلب" وهكذا يمكن التعرف على جميع العلاقات الواردة بين مجموعات هذا المخطط.

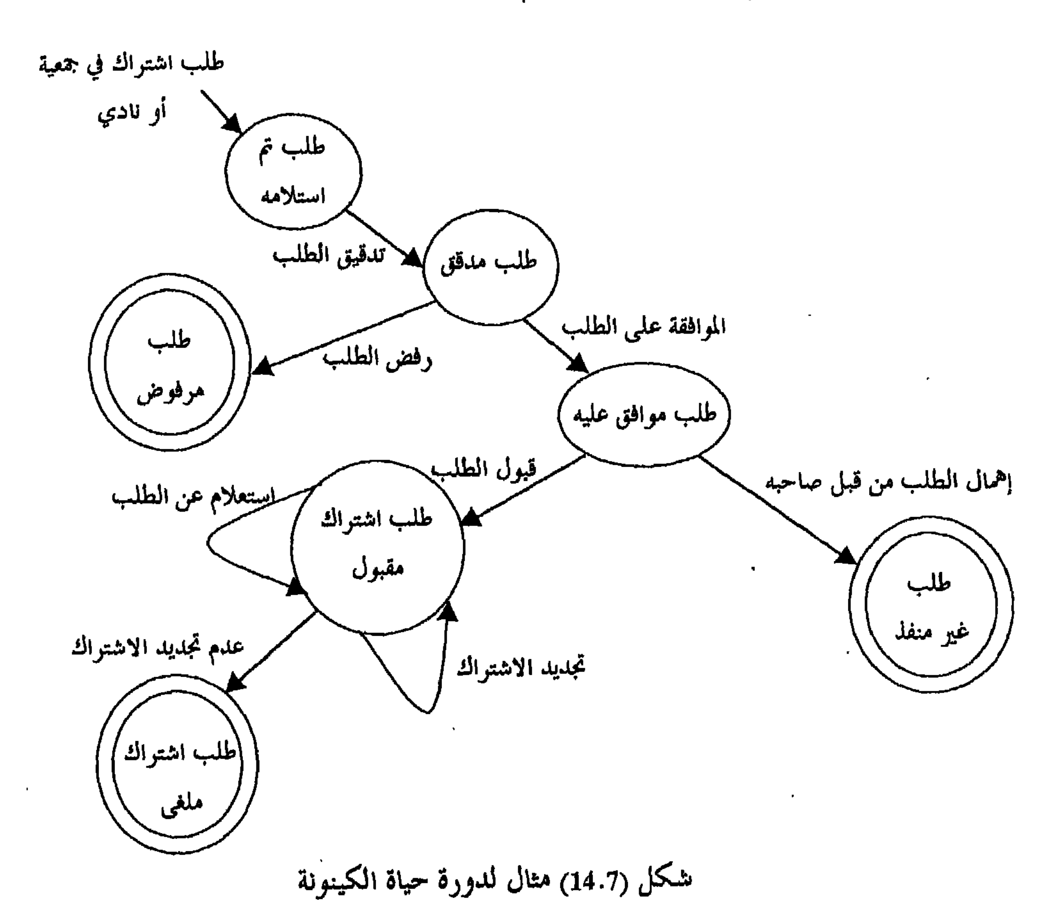
كما تجدر الإشارة إلى طريقة تمثيل الروابط Cardinality بين المجموعات في هذه المخططات، حيث تستخدم لذلك أسهم ذو اتجاهين بجانب المستطيلات التي تكتب فيها أسماء الروابط، ويعني هذا السهم وجود علاقة واحدة فقط (أي انه يوجد للطلب الواحد زبون واحد فقط)، أما عدم وجود السهم فيعني إمكانية وجود عدة روابط (مثلاً الزبون يمكن أن يتقدم بعدة طلبات) وهكذا.

وأخيراً نلاحظ دائرة فوق المستطيل الذي يحوي اسم العلاقة التي تصل بين " رقم الطلب" و"رمز المنتج"، ووجود رابطة تصل هذه الدائرة مع المجموعة المسماة " الكمية المطلوبة". وهذا يعني أن مجموعة" الكمية المطلوبة" ترتبط بآن واحد مع كل من مجموعة " رقم الطلب" ومجموعة " رمز المنتج" أي انه لا يمكن تحديد الكمية المطلوبة إلا بمعرفة كل من " رقم الطلب" و " رمز المنتج".

6)مخطط تاريخ حياة الكينونة Entity Life History Diagrams

يستخدم هذا النوع من المخططات لتتبع الحالات المختلفة التي يمكن أن تمر بها الكينونة منذ ظهورها، أو دخولها إلى النظام وحتى خروجها أو بقائها فيه بشكل دائم، ويتميز هذا النوع من المخططات بكونه يعرض التغيرات التي يمكن أن تحدث للكينونة خلال حياتها، بعكس المخططات التي درسناها سابقاً والتي تمثل بنية البيانات بشكل ثابت Static Structure.

فمن خلال هذه النماذج يمكن أن نتعرف على جميع الأحداث Events التي يمكن أن تتعرض لها الكينونة في النظام، فمن خلال هذه الأحداث تنتقل الكينونة من حالة State إلى أخرى. ويطلق على هذه العملية التحولات Transitions، وهكذا فإن مخطط تاريخ حياة الكينونة يعرض التحولات والحالات المختلفة التي يمكن أن تمر بها الكينونة ويتم تمثيل هذه الحالات باستخدام رمز الدائرة، أما التحولات فيرمز إليها بأسهم تصل بين هذه الحالات . كما تعطى الدوائر (الحالات) تسميات مختلفة للدلالة عليها ، وكذلك يتم تسمية التحولات وتكتب هذه التسميات ضمن الدوائر وفق الأسهم كما هو مبين في الشكل (14.7)



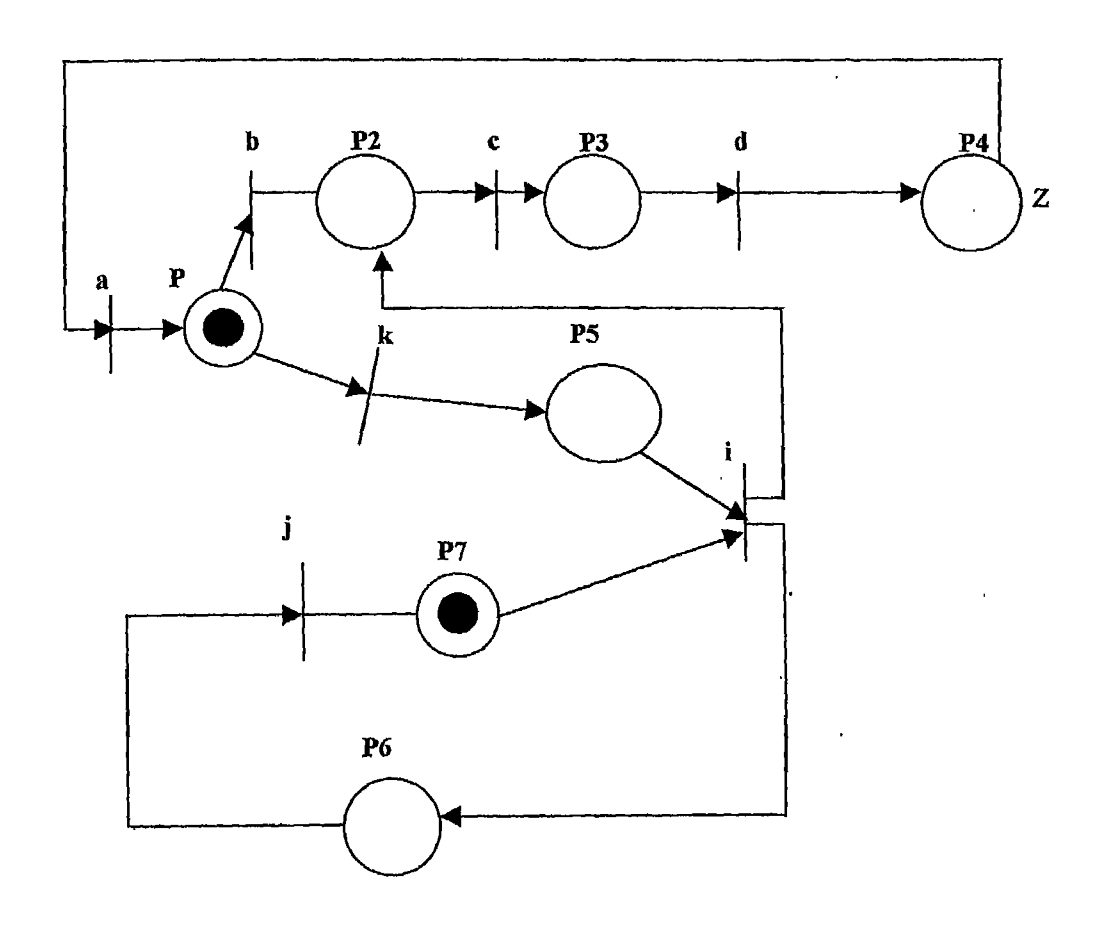
شكل (14.7) مثال لدورة حياة الكينونة

نلاحظ في الشكل (14.7) أعلاه وجود دوائر ذات خط مزدوج ، وهـــي تمثل تلك الحالات التي تصبح فيها الكينونة غير هامة للنظام ، فالطلبات المرفوضـــة وكذلك الطلبات التي يهملها أصحابها لن يكون لها أي قيمة بالنســـبة للنظـــام . ونلاحظ أيضا استخدام الأسماء لتسمية الحالات المختلفة للكينونة والأفعال لتسـمية التحولات الخاصة بها . كما نلاحظ أن بعض التحولات انطلقت مـــن العمليـــة وعادت إليها ، وهذا يعني إنها تحدث تغييرا في بعض خصائص الكينونة ولكـــن لا تحدث أي تغيير في حالتها ، فمثلا عندما يدفع المشترك قيمة الاشتراك السنوي يبقى طلبه ساري المفعول ، وإذا استعلم المشترك عن أي معلومات مثلا معرفـــة تـــاريخ تجديد الاشتراك والمبلغ المطلوب للسنة الحالية وغير ذلك ، فإن مثل هذه العمليـــات لا تسبب تغيير لحالة الطلب . وبشكل عام تستخدم هذه المخططـــات لتحديـــد عمليات النظام System Transactions حيث أن كل تحول من حالـــة إلى حالـــة يمـــثل إحدى العمليات التي يجب أن يقوم بها النظام . كما يمكن استخدام هـــذه المخططات لتدقيق مخططات تدفق البيانات والتأكد من صحتها ، فالعمليـــات في المخططات DFD تمثل التحولات التي تتضمنها مخططات دورة حياة الكينونة هذه .

7)شبكات بتري Petri Nets :

وهي أسلوب جديد لوصف التغيرات التي يمكن أن تحـــدث في النظـــام . تستخدم هذه الشبكات ثلاث عناصر لبناء نموذج النظام وهي :

- المكان Place : ويرمز إليه بدائرة .
- التحول Transition : ويرمز إليه بخط عامودي، ويمثل عمليات معينة.
- العلامة أو الرمز Token: وترسم بشكل نقطة سوداء، تتنقل ضمـــن الشبكة من مكان إلى آخر.



شكل (15.7) شبكة بتري

ويبين الشكل(15.7) التالي مثالا على هذا النوع من المخططات .

يتضح من المخطط أعلاه وجود سبعة أماكن يرمز إليها بحروف كبيرة مرقمة من P1 وحتى P7، ويربط بينها أيضا سبعة تحولات هي a،b ، c، d، k، i، j.

والقواعد التي تعمل بموجبها هذه الشبكات هي:

- توجد العلامة أو الرمز (Token) بشكل دائم في أحـــــد الأمكنـــة في الشبكة.
- يبدأ أو ينطلق التحول عندما توجد علامة أو رمز في جميـــع أمـــاكن الدخول إليه.
- عندما ينطلق التحول ، يضع العلامة في جميع أماكن الخروج منه .
- يمكن إستخدام العلامة الموجودة في مكان معين لإطلاق تحول واحـــد فقط .

نلاحظ في الشكل أعلاه وجود علامـــــات في الأمكنــــة P1,P7 فقـــط . والتحول i لا يمكن أن ينطلق نظراً لأنه لا توجد علامة إلا في واحد فقط من أماكن الدخول إليه وهو P7 . والـــعلامة الموجودة في المكان P1 يمكن استخدامها فقـــط لإطلاق التحول b أو k . فإذا انطلق التحول b فان المكان P2 سوف يحصل علـــى هذه العلامة التي ستنطلق بواسطة التحول c وتستقر في P3 ، الذي يطلقها بـــدوره خلال التحول d ثم a لتعود إلى المكان P1 ، ويمكن لهذه الدورة أن تكرر . ولكـــن إذا انطلق التحول k فيضع العلامة في المكان P5 وعندها يمكن إطلاق التحـــــول i الذي يضع العلامة في الأمكنة P2 و P6 وبنفس الطريقة تعود العلامة إلى الأمكنــــة التي كانت فيها في البداية وهي P1 و P7 . وهكذا يمكن تكرار هذه العملية عــــدة مرات .

تستخدم هذه الشبكات لنمذجة العمليات المسبقة التحديد (مثل إجـــراءات معالجة نموذج معين ضمن النظام) . فمثلاً يمكن أن يمثل المخطط أعـــلاه عمليـــات معالجة نموذج Form معين . وتمثل العلامة أو الرمز الموجود في مكان مـــا أن هـــذا النموذج تجري معالجته حالياً (تنفيذ عملية معينة فيه)، أما التحول فهو قرار يتعلـــق

بالنموذج ، فمثلا التحول b يمكن أن يعني أن النموذج تمت معالجته بصورة صحيحة ، والتحول k يدل على أن هذا النموذج تمت معالجته بطريقة غير صحيحة. فالنموذج الذي كانت معالجته صحيحة يمكن أن ينتقل إلى الأماكن P2 و P3 و P4 ثم يعود إلى P1، أما النموذج الذي تمت معالجته بشكل خاطئ فيصل إلى المكان P5 لتصحيحه قبل أن ينتقل إلى P2. أما المكان P6 فيمكن أن يمثل عدادا يعد مرات الخطأ التي يمكن أن تحدث في النظام .

أسئلة الفصل

1- عدد واشرح الخطوات الثلاث لنمذجة بيانات النظام ؟

2- ما هي أهم الأدوات المستخدمة لنمذجة البيانات ؟

3- اشرح مخططات هيكل البيانات : عناصرها وخطوات رسمها ؟

4- اشرح عناصر مخططات الكينونة – العلاقة ؟

5- اشرح قواعد رسم مخططات الكينونة – العلاقة ؟

6- ما هي الكينونات المرتبطة وما هي المجموعات الفرعية ومتى تستخدم كل منهما ؟

7- اشرح خطوات إعداد مخططات الكينونة – العلاقة ؟

8- قارن بين مخططات تدفق البيانات ومخططات الكينونة – العلاقة ؟

9- اشرح مخططات الكينونة وقارن بينها وبين مخططات الكينونة – العلاقة ؟

10- ما هي مخططات العلاقة الثنائية واشرح أهم عناصرها ؟

11- ما هي أهداف مخطط تاريخ حياة الكينونة وبم يتكون ؟

12- اشرح شبكات بتري : مكوناتها واستخداماتها ؟

تمارين

1- ارسم مخطط الكينونة- العلاقة لنظام شركة لصناعة الألبسة الجاهزة ، تم توصيفه على النحو التالي :

- تتكون الشركة من عمال يتم تعريفهم من خلال رمز العامل واسم العامل (الاسم الأول واسم العائلة) وتاريخ الولادة ، والمهنة .
- يكون العمال مسؤولين عن تنفيذ أوامر الإنتاج التي تتضمن : رقم أمر الإنتاج – تاريخ أمر الإنتاج – وصف المنتجات المطلوبة وكمياتها والأسعار المتفق عليها . علما أن أمر الإنتاج الواحد يعود إلى زبون واحد . ويمكن أن يكون العامل مسؤولا عن تنفيذ عدد من الأوامر.
- يتم جدوله تنفيذ الأوامر في الشركة بشكل سلسلة من الأعمال . والعامل المسؤول عن أمر الإنتاج يقدم طلبات رسمية للأقسام المختصة التي يمكن ان تقوم بهذه الأعمال (التفصيل – الخياطة – الكوي – التغليف).ويتم إعطاء هذه الطلبات أرقاما، وتحديد تاريخ بداية وتاريخ نهاية العمل لكل طلب من هذه الطلبات .
- في ضوء هذه الطلبات تقوم الأقسام (لكل قسم رمز واسم ومدير) بتحديد الأعمال المطلوبة منها . (إعطائها أرقاما) وتكلفة متوقعة.

- يتطلب تنفيذ كل عمل عدة أنواع من المواد ويتم تعريـــف المـــواد بواسطة رموزها وأسمائها .

2-ارسم مخطط الكينونة للحالة الواردة في التمرين السابق ؟

3- ارسم مخطط تاريخ حياة الكينونة لأمر الإنتاج ؟

الفصل الثامن

توصيف العمليات وقاموس البيانات

بعد إعداد مخططات تدفق البيانات ومخطط الكينونة – العلاقة ، يقوم المحلل بتوصيف العمليات الموجودة في مخططات التدفق (PD) Process Description ، كذلك إعداد قاموس البيانات (DD) Data Dictionary لضمان توحيد التسميات وتوصيفها وتوثيقها بشكل منسق ومنسجم . وتمثل هذه الوثائق جميعاً نموذج النظام System Models الذي يتكون من الوثائق التالية:

- مخططات تدفق البيانات في النظام DFDs.
- مخطط العلاقة الكينونة E-RD
- توصيف العمليات PDs.
- قاموس البيانات DD.

وسندرس في هذا الفصل طرق وأساليب توصيف العمليات وإعداد قاموس بيانات النظام.

1– توصيف العمليات : Process Description

إن جميع العمليات الموجودة في مخططات تدفق البيانات يجب توصيفها ، أي شرح الخطوات اللازمة لأداء هذه العمليات . وهذا يعني تحديد كيفية تحويل التدفقات الداخلة إليها إلى التدفقات الخارجة منها.

ونظراً لأن مخططات تدفق البيانات تتكون من عـــدة مســتويات، فــإن العمليات الموجودة في كل مستوى يجب أن توصف بطريقة تختلف عن المســتويات الأخرى. فالعمليات الموجودة في المخطط العام لتدفق البيانـــات Top-Level DFD يمكن توصيفها باللغات الطبيعية لكونها ذات طبيعة إجمالية وموجزه. أما العمليـــات الموجودة في المخططات التفصيلية أي في المستوى الأدنى فيجب إليها يتم توصيفـــها بطريقة واضحة ومحددة لا لبس فيها ولا غموض، ويجب إليها يكون هذا التوصيـــف سهل الفهم لكل المشاركين في عملية التحليل. ولقد تطورت خلال السنوات الماضية أساليب وأدوات عديدة لتوصيف هذه العمليات مثل اللغات البنيويـــة وجـــداول القرار وشجرات القرار وغيرها. وسوف نتعرف فيما يلـــي علـــى أهـــم الأدوات المستخدمة في توصيف العمليات.

1.1 اللغة البنيوية Structured Language

تستخدم التراكيب اللغوية البنيوية في توصيف العمليات ، وخاصة تلـــك الموجودة في المستوى الأدنى لمخططات تدفق البيانات، ونظراً لان اللغة الإنجليزية هي الأكثر انتشاراً في هذا المجال فإنه يتم استخدام تعابـــير هـــذه اللغـــة في عمليـــات التوصيف. ويتم توصيف العمليات ، باستخدام هذه اللغات البنيوية ، من خـــلال تشكيله من البنى التالية: سلاسل من الجمـــل الأمريـــة Imperative Sentences والقرارات Decisionsوالدورات أو التكرارات.

أ) الجمل الأمرية Imperative Sentences

تتكون هذه الجمل غالبا من فعل أمر لتنفيذ عمل ما يليه مباشـــرة أسمـــاء البيانات موضوع هذا الأمر ، أي التي ستجري عليها العمليات المطلوبة بموجب هذا الأمر . فمثلا الجمله : Subtract Tax From Total يقصد بها اطرح الضريبة مـــن

الإجمالي. ومن المهم عند صياغة هذه الجمل إليها يكون الفعل المستخدم واضح المعنى مثل Add أو Compute ، ويجب الابتعاد عن الأفعال ذات المعـــاني المبهمـــة مثـــل Process أو Handle أو Edit أو غيرها.

كما يمكن أن تتضمن هذه الجمل العمليات الحسابية والمنطقية . ويجـــب أن تكتب أسماء البيانات والملفات بحروف كبيرة، كما هي واردة في مخططات التدفـــق وفي قاموس البيانات . وتستخدم اللغة البنيوية الكلمات المفتاحيه التاليـــة لتجميـــع الجمل الأمرية في كتل Blocks، وفي تحديد تفريعات القرارات و الدورات . وهـذه الكلمات هي:

BEGIN	**REPEATE**	**IF**
END	**UNTIL**	**THEN**
CASE	**DO**	**ELSE**
OF	**WHILE**	**FOR**

ب) القرارات : Descisions

يمكن استخدام نوعان من بنى القرارات في توصيف العمليـــــات بواســـطة اللغـــة البنيوية:

- بنية التفرع وهي: (مجموعة جمل أمر 1) Then شرط IF

(مجموعة جمل أمر 2) ELSE

- بنية الشروط المتعددة CASE:

وفيها يتم إجراء اختبار قيمة متغير معين ثم يتم اختيار واحد من عدد مـــن التفريعات الممكنة في ضوء نتيجة هذا الاختبار.

ويمكن استخدام أسلوب القرارات المتداخلة لتوصيف العمليات بشرط إليها لا يتعارض ذلك مع متطلبات سهولة الفهم.

جـ) التكرارات Repetition

يتم توصيف التكرارات أيضاً بواسطة اللغة البنيوية باستخدام البنى التالية:

-التكرار ما دام الشرط محققاً:

```
WHILE شرط DO
        BEGIN
        مجموعة جمل أمريه
        END
```

– التكرار حتى يتحقق الشرط:

```
REPEATE
        BEGIN
        مجموعة جمل أمريه
        END
UNTIL شرط
```

مثال:

يبين الشكل (1.8) توصيف عملية طباعة تقرير يحتوي معلومات تفصيلية عن الزبائن.

الشكل (1.8) توصيف العمليات (مثال)

```
Initialize totals to zero.
Initialize the page number to 1.
Get the report date from the system
Set lines counter to zero .
Read a record from Customers Master file
While (not end of customers .master file)
        IF (lines printed < = 0 ) THEN
        Advance paper to top of page
        print heading lines
        Add 1 to page number
        Set lines counter to 50
END-IF
Print Detail line
Add YTD-SALES , YTD-PAYMENTS
                To GRAND-TOTALS
Subtract 1 from lines-counter
Read a record from customer-Master file
End-While
PRINT grand total line three lines after last detail line
```

2.1 جداول القرارات Descision Tables

يمكن باستخدام اللغة البنيوية توصيف جميع العمليات . ولكن عندما تكون العملية معقده وتتضمن شروط كثيرة ، فإن توصيفها باستخدام اللغة البنيوية يكون معقداً وصعب الفهم . لذلك يستخدم لمثل هذه الحالات المعقدة أسلوب جداول القرارات . وتعتبر هذه الجداول أسلوبا سهل الفهم ويمكن من خلاله توضيح وشرح العمليات المعقدة والتي تتطلب شروطاً عدديه ومركبه.

يتألف جدول القرارات من قسمين هما قسم الشروط وقسم الأفعال ، وينقسم كلاً من هذين القسمين بدورهما إلى قسمين هما على التتالي قائمة الشروط وقائمة تدوين الشروط ، وكذلك قائمة الأفعال وقائمة تدوين الأفعال وذلك على النحو المبين في الشكل (2.8) . أما الشكل (3.8) فيبين مثالاً لجداول قرار يتعلق بعمليه قبول أو رفض طلبات الزبائن

	قواعد القرار
قائمة الشروط (تكتب هنا جميع الشروط المتعلقة بالقرار).	**أعمدة الشروط** (يمثل كل عامود التركيب المختلفة لحدوث الشروط).
قائمة الأفعال (تكتب هنا جميع الأفعال الممكنة).	**أعمدة الأفعال** (يمثل كل عامود الأفعال المطلوبة في ضوء تحقيق تشكيلية الشروط في ذلك العامود).

شكل (2.8) الأقسام الاربعه لجداول القرارات

جدول قرار عملية قبول أو رفض الطلبات	قواعد القرار			
رصيد الزبون أقل من حد الائتمان المسموح به	نعم	نعم	نعم	لا
الزبون مصنف بكونه يدفع بانتظام	نعم	لا	لا	
الزبون لديه موافقة استثنائية من الإدارة		نعم	لا	
قبول الطلب	X	X		
رفض الطلب			X	X

شكل (3.8) جدول قرارات (مثال)

نلاحظ من الجدول (3.8) وجود أربع قواعد لاتخاذ القرار في هذه العمليـة ويتم قراءتها على النحو التالي:

1) إذا كان رصيد الزبون أقل من حد الإئتمان المسموح به وكـان الزبـون مصنفا بكونه يلتزم بالدفع بانتظام فاقبل الطلب.

2) إذا كان رصيد الزبون اقل من حد الإئتمان المسموح به وكان الزبـون لا يتصف بكونه يدفع بانتظام وكانت لديه موافقـة مـن الإدارة فـاقبل الطلب.

3) إذا كان رصيد الزبون أقل من حد الإئتمان المسموح به وكان الزبـون لا يتصف بكونه يدفع بانتظام وليس لديه موافقـة اسـتثنائية مـن الإدارة فـأرفض الطلب.

4) إذا كان الزبون قد تجاوز حد الإئتمان المسموح به فارفض الطلب.

خطوات إعداد جداول القرارات:

يمكن إعداد جداول القرارات باتباع سلسلة من الخطوات سنقوم بدراستها من خلال المثال التالي : لنفترض وجود الوصف التالي لعملية اتخاذ القرار المتعلقة بمعالجة طلبات صرف المواد من المخازن:

" عندما تكون الكمية المطلوبة من مادة ما ضمن الحدود المسموح بها، ويكون الطلب موافقاً علية من قبل الإدارة ، فاطرح الكمية المطلوبة من رصيد المادة، أضفها إلى حقل الكميات المصروفة ، وقم بإعداد مذكرة التسليم اللازمة لتسليم المادة إلى الجهة الطالبة وإذا كان الرصيد الجديد أقل من مستوى إعادة الطلب يجب إصدار طلب شراء . ومن الطبيعي أن تتوفر الكمية المطلوبة في المستودع لتتم تلبية هذا الطلب . يجب رفض الطلب عندما تكون الكمية المطلوبة أكبر من الحدود المسموح بها ، وكذلك إذا لم يحمل الطلب موافقة الإدارة. من الواضح في هذه الحالة البسيطة بأن الوصف العادي باستخدام اللغة البنيوية سيجعل فهمها صعباً . ولذلك سنستخدم جداول القرارات لوصف هذه الحالة بطريقة أكثر وضوحاً وتحديداً من خلال اتباع الخطوات التالية:

1. تحديد الشروط: Define Conditions

في هذه الخطوات يتم دراسة الوصف الوارد أعلاه لاستخراج قائمتين الأولى بالشروط Conditions والثانية بالأفعال Actions الناتجة عنها . فإذا تأملنا النص أعلاه نجد انه يحتوي الشروط التالية:

C1: الكمية المطلوبة اقل أو تساوي الحدود المسموحة للطلب.

C2: الطلب يحمل تأشيرة الموافقة من الإدارة.

C3: الكمية المطلوبة متوفرة في المستودع.

C4: الرصيد الجديد للمادة اقل أو يساوي مستوى إعادة الطلب.

C5: الكمية المطلوبة أكبر من الحدود المسموحة للطلب.

C6: الطلب لا يحمل تأشيرة الموافقة من الإدارة.

ولدى دراسة الشروط الواردة أعلاه نجد إليها الشرط C5هو معاكس تماماً للشرط C1. وكذلك الشرط C6فهو عكس الشرط C2. ولذلك يتم حذف الشرطين C5 و C6 لكونهما يعتبران تكرار غير ضروري وبذلك يصبح لدينا أربعة شروط فقط هي:

C1: الكمية المطلوبة اقل أو تساوي الحدود المسموحة للطلب.

C2: الطلب يحمل تأشيرة الموافقة من الإدارة.

C3: الكمية المطلوبة متوفرة في المستودع.

C4: الرصيد الجديد للمادة اقل أو يساوي مستوى إعادة الطلب.

2) تحديد الأفعال الممكنة Define Actions

من دراسة التوصيف المذكور أعلاه لهذه المشكلة يمكن تحديد قائمة الأفعال على النحو التالي:

A1 اطرح الكمية المطلوبة من رصيد المادة.

A2 أضف الكمية المطلوبة إلى حقل الكميات المصروفة.

A3 قم بإعداد مذكرة التسليم..

A4 أصدر طلب شراء للمادة .

A5 ارفض الطلب.

3) رسم الجدول الفارغ:

نرسم في هذه الخطوة جدولاً فارغاً نحدد عدد اسطره وعدد أعمدته على النحو التالي:

- عدد الأسطر = عدد الشروط + 1 = 5 اسطر.
- عدد الأعمدة = جداء القيم الممكنة للشروط.

وهذا يعني انه لدينا 4 شروط لكل منها قيمتان (نعم و لا) فيكون الجداء 2× 2× 2×2 = 16 عاموداً. أضف إليها عاموداً آخر لتدوين أسماء الشروط. وهكذا نرسم جدولاً فارغاً يحوي خمسة اسطر وستة عشر عاموداً، ونكتب فيه الشروط وأرقام الأعمدة ، كما هو مبين في الشكل (4.8).

الشروط	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
الكمية المطلوبة ضمن الحدود المسموحة																
الطلب موافق عليه من قبل الإدارة																
الكمية المطلوبة متوفرة في المستودع																
الرصيد الجديد اقل أو يساوي إعادة الطلب																

4) كتابة تدوينات الشروط في الجدول:

وذلك على النحو التالي :

- نقسم عدد الأعمدة على عدد القيم الممكنة للشرط الأول (2) فتحصل على معامل التكرار الذي يدل على عدد المرات التي يجب فيها كتابة كل قيمة في هذا السطر . وهكذا فعند قسمة 16 ÷ 2 = 8 وهكذا نكتب في السطر الأول حرف Yللدلالة على تحقيق الشرط ثمانية مرات، ونكتب N ثمانية مرات أخرى للدلالة على عدم تحقيق الشرط.
- نكرر العملية السابقة بنفس الطريقة لكل سطر من اسطر الشروط الأخرى ولكن تحديد معامل التكرار في كل سطر يكون بقسمة معامل التكرار في السطر السابق على عدد القيم الممكنة للشرط الموجود في هذا السطر ، وهكذا فإن معامل تكرار السطر الثاني هو 8÷2=4 فتحصل على الجدول المبين في الشكل (5.8).

الجدول (5.8)

الشروط	1	2	3	4	5	6	7	8	9	.1	11	12	13	14	15	16
الكمية المطلوبة ضمن الحدود المسموحة	Y	Y	Y	Y	Y	Y	Y	Y	Y	N	N	N	N	N	N	N
الطلب موافق عليه من قبل الإدارة	Y	Y	Y	Y	N	N	N	N	N	Y	Y	Y	N	N	N	N
الكمية المطلوبة متوفرة في المستودع	Y	Y	N	N	Y	Y	N	N	Y	Y	N	N	Y	Y	N	N
الرصيد الجديد أقل أو يساوي مستوى إعادة الطلب	Y	N	Y	N	Y	N	Y	N	Y	N	Y	N	Y	N	Y	N

شكل (5.7) كتابة تدوينات الشروط في الجدول

5) **تكثيف أو اختصار جداول القرارات :** نلاحظ أن جدول القرارات الذي نقوم باعداده يتضمن الكثير من التكرارات التي لا فائدة منها ، ولذلك يجب اختصاره قدر الإمكان وتبسيطه ليكون مفهوما .ويتم هذا التبسيط بحذف قواعد القرار التي لا معنى لها ؛ فمثلا القواعد من 9 حتى 16 فانه مهما كانت قيم الشروط الأخرى فإنها لا تعني شيئا لان الشرط الأول غير محقق وبالتالي سيرفض الطلب في جميع الأحوال ولذلك يمكن حذف الأعمدة من 10 حتى 16 وكذلك الأمر حيث يمكن حذف الأعمدة 6 و 7 و 8 لأنه طالما لا توجد موفقه من قبل الإدارة فسيرفض الطلب مهما كانت قيمة الشروط الأخرى.

وهذا يصبح جدول القرارات بعد اختصاره متضمنا خمس قواعد فقط كما هو مبين في الشكل (6.8).

5	4	3	2	1	
N	Y	Y	Y	Y	الكمية المطلوبة ضمن الحدود المسموحة
	N	Y	Y	Y	الطلب موافق عليه من قبل الإدارة
		N	Y	Y	الكمية المطلوبة متوفرة
			N	Y	الرصيد الجديد أقل أو يساوي مستوى إعادة الطلب

شكل (6.8) الشكل المختصر لجدول القرارات

6)إضافة الأفعال Action:

في هذه الخطوة يتم إضافة عدد من الأسطر إلى الجدول يساوي عدد الأفعال التي يتم تحديدها في الخطوة الثانية ونكتب أسماء هذه الأفعال تحت أسماء الشرط مباشرة.

أما تدوينات الأفعال فيتم بكتابة الرمز X أو ✓ في الخلايا الموجودة على تقاطع الفعل مع العامود الذي يمثل تدوينات الشروط الخاصة بهذه القاعدة . ويبين الشكل (7.8) الشكل النهائي لجدول القررارات .

	1	2	3	4	5
الكمية المطلوبة ضمن جدول الطلب	Y	Y	Y	Y	N
الطلب موافق عليه من الإدارة	Y	Y	Y	N	
الكمية المطلوبة متوفرة في المستودع	Y	Y	N		
الرصيد الجديد أقل أو يساوي حد إعادة الطلب	Y	N			
اطرح الكمية المطلوبة من رصيد المادة	✓	✓			
أضف الكمية إلى حقل الكميات المصروفة	✓	✓			
قم بإعداد مذكرة التسليم	✓	✓			
أصدر طلب شراء للمادة	✓		✓		
ارفض الطلب			✓	✓	✓

شكل (7.8) الشكل النهائي لجدول القرارات.

2- قاموس البيانات : Data Dictionary

يهدف قاموس بيانات النظام إلى تنميط جميع التسميات المستخدمة في النظام وإعطاء توصيف موحد لها ليكون مرجعا أساسيا من مراجع النظام . فعندما نرغب الحصول على تفاصيل تتعلق بأي تدفق بيانات أو مخزن بيانات أو عملية في مخططات

التدفق DFD أو مخططات الكينونة العلاقة E-RD، يمكن الرجوع إلى قاموس بيانات النظام للحصول على هذه التفاصيل.

إن قاموس البيانات هو عبارة عن أداة (وثيقة) يتم من خلالها تعريف وتوصيف مخازن وتدفقات البيانات ، وكذلك العمليات الموجودة في مخططات النظام . ويجب أن تكون تسميات هذه التدفقات في مخططات النظام منسجمة مع التسميات الموجودة في القاموس . وهكذا يمكن تعريف قاموس البيانات بأنه طريقة لترتيب وتسجيل فقرات البيانات المستخدمة في النظام ، يتم من خلالها تحديد أسماء هذه البيانات وأشكالها وحجومها وبنيتها.

والقاموس هو عبارة عن سلسلة من القيود المختلفة تسرد بشكل متسلسل أبجديا جميع فقرات النظام وهذا يشمل :

- تدفقات البيانات Data Flows
- مخازن البيانات Data Stores
- عناصر البيانات Data Elements
- كما يمكن أن يحتوي قاموس البيانات توصيف العمليات الموجودة في المستويات العليا في مخططات تدفق البيانات.

تكتب أسماء البيانات في القاموس بنفس الطريقة المستخدمة في المخططات أي بحروف إنجليزية كبيرة ويمكن استخدام الرمز "–" للفصل بين الكلمات لتسهيل فهم وقراءة هذه الأسماء ، التي يفضل أن تعبر بشكل واضح عن مضمون هذه الفقرات.

1.2 الرموز المستخدمة:

يتم توصيف هياكل البيانات عادة في القاموس باستخدام رموز خاصة يمكن التعرف عليها من خلال المثال التالي :

VENDO-INVOICE = INVOICE–NO + VENDOR NAME + INOICE-AMMOUNT + INVOICE–DUE–DATE + [VENDOR-PHONE | VENDOR–FAX] + (SHIPPING–DATE) + $_{1}^{20}${INVOICE–ITEM- LINE}

* على الأقل يجب استلام نسختين من كل فاتورة *

نلاحظ من القيد أعلاه أن بنية تدفق البيانات المسماة VENDOR INVOICEتتضمن خمسة عناصر بيانات ، أحدها اختياري ، ويمكن أن تحتوي حتى عشرين تدفقاً يخزن فيها البيانات التفصيلية للفاتورة . ونلاحظ استخدام الرموز التالية في صياغة التوصيف الوارد أعلاه :

الرمز	المعنى
=	يكافئ
+	و
ملاحظة *	تكتب الملاحظات بين نجمتين
(مكون)	مكون اختياري
{مكون}	مكون يمكن أن يأخذ قيما متعدده من 1 حتى N مرة
[مكون 1 / مكون 2]	يستخدم فقط مكون واحد من المكونات الواردة في القائمة

أما أهم قواعد صياغة تسميات قاموس البيانات فهي:

- استخدام أسماء ذات دلالة واضحة، وفريدة غير متكررة.
- استخدام أسماء رديفة Aliasis إذا دعت الحاجة .
- استخدام أسماء مركبة للدلالة على مجموعة عناصر بيانات بدلا من سرد أسماء هذه العناصر. ووضع قيود خاصة في القاموس لكل مكون مركب تسرد أسماء العناصر المكونة لها.

2.2 توصيف أنواع التدفقات المختلفة:

يتضمن قاموس البيانات قيداً خاصاً لكل نوع من أنواع التدفقات المختلفة للبيانات في النظام ، وتختلف هذه القيود من نوع بيانات إلى آخر . وسنتعرف فيما يلي على طريقة التوصيف المستخدمة لكل نوع من هذه القيود.

أ) توصيف عناصر البيانات:

عنصر البيانات Data Elements هو أبسط صيغة في القاموس ويمثل العنصر عادة حقل بيانات واحد مثل رمز المنتج أو اسمه ويحتوي توصيف عناصر البيانات على المعلومات التالية:

- اسم عنصر البيانات: مثلاً PRODUCT - ID
- الأسماء المستعارة أو الرديفة: مثلاً PRODUCT-NO
- وصف العنصر: مثلاً يتكون من ثمانية محارف تمثل الثلاث الأولى منها رمز الصنف التي ينتمي إليه المنتج.

- المجال: ويقصد به مجال القيم التي يمكن أن يأخذها هذا العنصر.
- مكان الاستخدام : أسماء السجلات والتدفقات المركبة التي يمكن أن تحتـــوي هذا العنصر .

ب) توصيف هياكل البيانات:

يقصد بهياكل البيانات مجموعة مركبة من عناصر البيانات تظــهر معــاً في أماكن مختلفة في مخططات البيانات . ويتم توصيف هياكل البيانات مـــن خـــلال العناصر التالية:

- اسم هيكل البيانات.
- بنية هيكل البيانات : ويمكن وصفها بالطريقة التي تعرفنا عليها في الفقرة السابقة، كما يمكن توصيف هذه البنية بطريقة هرمية على النحو التالي:

```
INVOICE
    INVOICE HEADING
        SUPPLIER
        ORDER NO
        ORDER DATE
    INVOICE LINE*
        PRODUCT CODE
        QUANTITY
        PRICE
    SUPPLIER PHONE FAX
```

- الوصف المادي لهيكل البيانات : مثلا نموذج قياسي مأخوذ من فواتير المورد.

- مكان الاستخدام : أسماء العمليات التي تستخدم هذا التدفق.
- مخازن البيانات : أسماء مخازن البيانات التي يمكن أن يخزن فيها.

جـ) توصيف تدفقات البيانات:

يمكن إليها يتكون تدفق البيانات من عنصر أو مجموعة عناصر بيانات وكذلك يمكن أن يحتوي هياكل بيانات . ويتم عادة توصيف التدفقات على النحو التالي:

- اسم التدفق
- مصدر التدفق : اسم العملية أو الكينونة الخارجية التي يأتي منها التدفق.
- وجهة التدفق : اسم العملية أو الكينونة الخارجية التي يتجه نحوها هذا التدفق.
- بنية التدفق : ويتم هنا سرد مكونات التدفق بالطريقة المستخدمة في توصيف هياكل البيانات.
- الحجم : مثلاً عدد التدفقات باليوم أو الساعة.
- الوصف المادي : مثلاً فاتورة ورقية أو إلكترونية.
- ملاحظات : يمكن هنا ذكر أية معلومات لا تندرج ضمن الفقرات أعلاه.

حـ) توصيف مخازن البيانات:

يتم توصيف مخازن البيانات من خلال الفقرات التالية:

- اسم مخزن البيانات: مثلاً الفواتير المدفوعة.
- توصيف المحتوى : مثلاً PAID—INVOICES + DATE .
- العمليات التي تستخدم المخزن : (أرقامها وأسماءها)
- الوصف المادي : مثلاً ملف يدوي أو ملف محوسب .

- الحجم : مثلاً حوالي 10000 سجل.

د) توصيف العمليات في قاموس البيانات:

يمكن أن يتضمن قاموس البيانات قيــــود أخـــرى لتوصيـــف العمليـــات والوحدات الوظيفية Modules بشكل موجز ، أما أهم فقرات توصيف العمليـــات فهي:

- رقم العملية.
- اسم العملية.
- التدفقات الواردة إليها.
- التدفقات الخارجة منها.
- مخازن البيانات المستخدمة من قبل العملية.
- وصف مختصر للعملية .
- طريقة التنفيذ (بشكل موجز أيضاً).

وتجدر أخيراً الإشارة إلى أهمية استخدام الأدوات المحوسبة لإنشـــــاء هـــذا القاموس (أدوات هندسة البرمجيات بمساعدة الحاسوب) حيث توفر للمحلل المزايا التالية:

- استخراج قائمة أبجدية بجميع التعاريف الموجودة في القاموس.
- التدقيق الآلي للتسميات والتعاريف والتأكد من انسجاميتها واكتمالها.
- إنشاء الإشارات المرجعية بين فقرات البيانات Cross references.

أسئلة الفصل

1. ما المقصود بتوصيف العمليات وما هي أهم الأدوات المستخدمة وما
2. ما هي اللغات البنيوية وما هي التراكيب المستخدمة فيها لتوصيف عمليات النظام ؟
3. ما هي مزايا جداول القرارات وما هي الأقسام التي تتكون منها هذه الجداول ؟
4. إشرح خطوات إعداد جداول القرارات ؟
5. ما هو قاموس البيانات وما هي الرموز المستخدمة فيه ؟ وما هي أهمية استخدامه؟
6. حدد المعلومات التي يجب تدوينها في قاموس البيانات لتوصيف أنواع التدفقات التالية:

- عناصر البيانات.
- هياكل البيانات.
- تدفقات البيانات.
- مخازن البيانات.
- عمليات النظام.

تماريـــن

1. صِف فاتورة الكهرباء التي تصلك كل شهر بشكل مقطع من قاموس البيانات ؟
2. صِف بيانات الهوية الشخصية أو جواز السفر بشكل قيود قاموس بيانات ؟
3. ارسم جدول القرارات الذي يمثل الحالة التالية:

تعطي إحدى الشركات لزبائنها حسماً على القيمة الاجمالية للفاتورة إذا تم تســـــديدها خلال عشرة أيام من استلامها ، وتحدد نسبة الحسم في ضوء ما يلي:

- إذا كانت قيمة الفاتورة أقل من 1200 دينار تكون نسبة الحسم 3% فقط
- إذا كانت قيمة الفاتورة بين 1200-3000 دينار فإن نسبة الحسم هي 7%.
- إذا كانت قيمة الفاتورة أكثر من 3000 دينار فإن نسبة الحسم 10 %.

علماً أن الطلبات المستعجلة لا تعطى أية حسميات .

4. ارسم جدول اتخاذ قرارات لعملية احتساب قيمة المياه المستهلكة من قبل المشـتركين، والذي يمثل الحالة التالية:

- إذا كان الاستهلاك الشهري حتى 50 متراً مكعباً فيكون سعر المـتر 0.5 دينار.
- إذا كان الاستهلاك الشهري من 50 —100 متراً مكعباً فإن سعر المـــتر يكون 75 قرشاً.
- إذا تجاوز الاستهلاك الشهري 100 متراً مكعباً فإن سعر المـــتر يصبـــح ديناراً واحداً.
- إذا كان الاشتراك للاستهلاك الصناعي يكون سعر المتر أعلـــــى بمـــره ونصف.
- إذا كان الاشتراك للإستهلاك التجاري يصبح سعر المتر ضعف ســـــعر الاستهلاك المترلي.

الفصل التاسع

نمذجة الأنظمة System modeling

1- مقدمة

لقد تعرفنا في .الفصول السابقة إلى عدد من الأدوات المستخدمة في تحليـــل الأنظمة كمخططات تدفق البيانات DFD، ومخططات الكينونة – العلاقـــة E-RD وتوصيف البيانات وقواميس البيانات . وتستخدم الأدوات المذكورة أعـــلاه لبنـــاء نماذج الأنظمة التي تجري دراستها . ونموذج النظام System Model هـــو شـــكل تجريدي للنظام الذي تجري دراسته يركز على المكونات المفاهيميـــة Conceptual Components له.

يستخدم النموذج للمساعدة في فهم وتحديد وظائف النظام الحالي وعملياتـه، وكذلك لتحديد وظائف النظام الجديد الذي سيحل محل النظام الحالي في المستقبل.

وبذلك يمكن تعريف نموذج النظام بأنه عبارة عن صورة أو وصف تجريدي للنظام يستخدم كأداة أو وسيلة لأغراض ثلاثة هي :

- لفهم وظائف النظام ومكوناته وعلاقاتها المختلفة .
- لاختبار الحلول المختلفة المتعلقة بتحسين أداء وكفاءة عمليات النظام .
- للاتصال بين جميع المهتمين والمشاركين بعملية تطوير النظام .

2- أهمية عملية النمذجة ومكوناتها :

أما النمذجة Modeling فنقصد بها عملية بناء النماذج التي تصف النظام أو النظم التي تجري دراستها . وتستخدم النمذجة في مرحلـــتي التحليـــل والتصميـــم

باعتبارها طريقة منتظمة Systematic تساعد في التوصل إلى نموذج النظام الجديد المقترح انطلاقا من النظام الحالي.

تستخدم عملية النمذجة المفاهيم المتعلقة بالنماذج المادية والنماذج المنطقية لأنظمة المعلومات. وبالرغم من أنه سبقت الإشارة إلى ذلك في الفصول السابقة إلا انه من المفيد التذكير بها هنا مرة أخرى.

يمثل **النموذج المادي للنظام** Physical System model الشكل المادي والواقعي للنظام الذي تجري دراسته وطريقة تنفيذ عملياته ووظائفه المختلفة.

فمثلاً عملية استلام الطلبات من الزبائن يمكن أن تتم بعدة طرق: يحضر الزبون ويقدم الطلب يدوياً، أو يرسله بالفاكس أو عبر الهاتف أو عبر البريد الالكتروني. والنموذج المادي يبين هذه العملية أي استلام الطلبات وطريقة تنفيذها.

أما **النموذج المنطقي** Logical System model فهو يركز على تبيان الأنشطة أو العمليات الجوهرية Essential Processes في النظام، ويقصد بها تلك العمليات أو الوظائف التي تمثل أساس وجود النظام، كعملية البيع مثلا أو عملية استلام الطلبات أو غيرها هي عمليات جوهرية ستبقى في النظام الجديد مهما كان مستوى حداثته وتطوره. وبشكل عام يمكن القول أن العمليات الأساسية أو الجوهرية هي تلك العمليات التي يجب أن يقوم بها النظام في جميع الأحوال ومهما كان مستوى التقنية المستخدمة فيه.

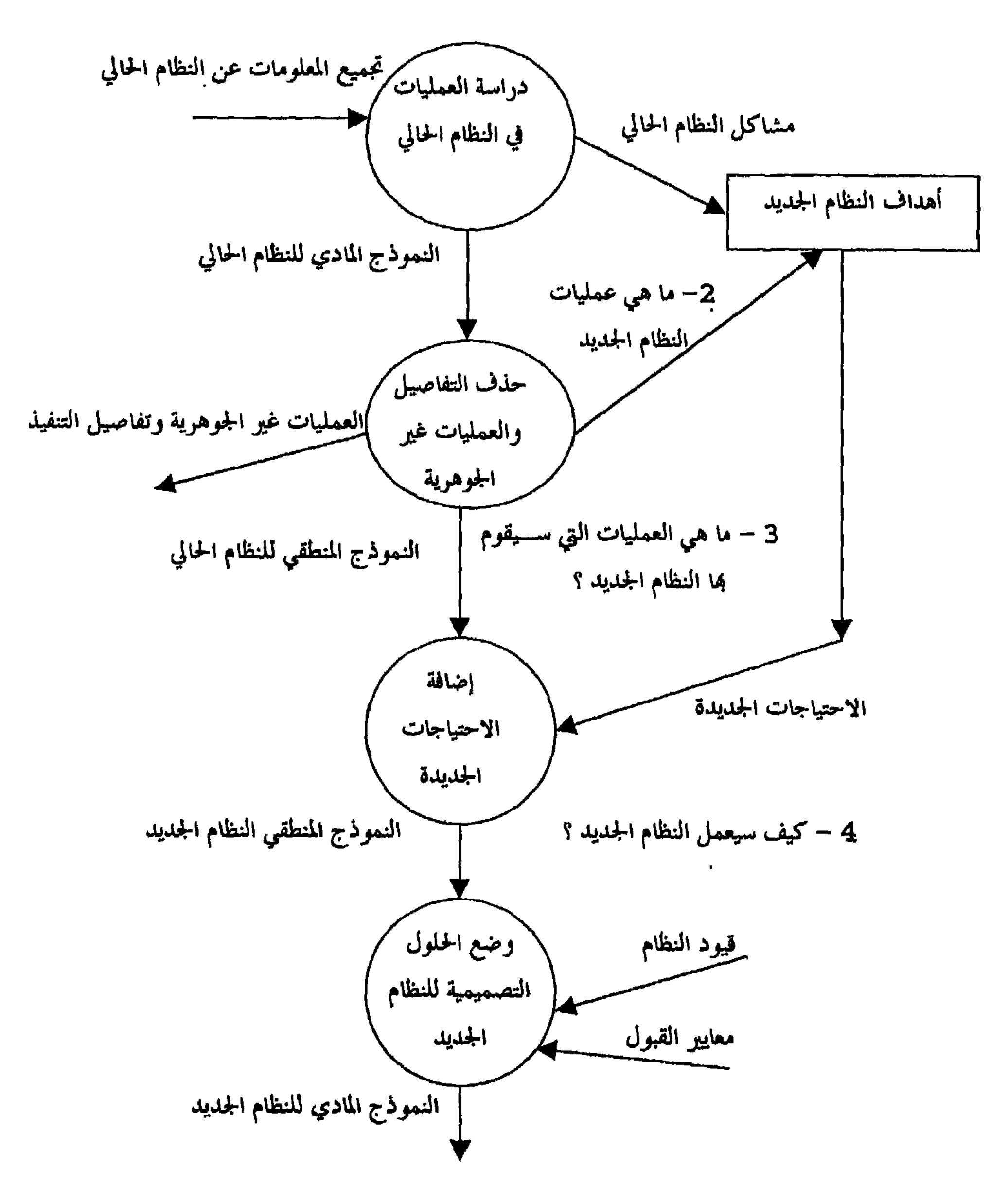
ويتم إعداد نماذج النظام خلال مرحلتي التحليل والتصميم بشكل نماذج بيانية Graphical Models أي بشكل رسوم تخطيطية تمثل مكونات النظام: عملياته ومدخلاته ومخرجاته ونقاط تجميع البيانات فيه وغيرها.

وتتميز هذه المخططات أو النماذج البيانية بسهولة فهمها من قبل جميع المشاركين في تحليل النظام، ولكنها لا تقدم معلومات تفصيلية عن المكونات المختلفة للنظام، لذلك يتم دعمها (أو إلحاق بها) بنماذج وصفيه Narrative Models تتضمن المعلومات التفصيلية للفقرات المختلفة للنظام. وهكذا فان مخططات تدفق البيانات ومخططات الكينونة – العلاقة هي مخططات بيانية تصف النظام بشكل إجمالي: مكوناته ومدخلاته ومخرجاته وفي عدة مستويات، ولكن المعلومات التفصيلية عن العمليات ووصف البيانات لا نجدها في هذه المخططات، بل في أدوات النمذجة الوصفية كقواميس البيانات وتوصيف العمليات، ولذلك فان نموذج نظام المعلومات سواء كان مادياً أو منطقياً فانه يتكون من الوثائق الأربعة التالية:

- مخططات تدفق البيانات DFDs
- مخطط الكينونة – العلاقة ERD
- توصيف العمليات
- قاموس البيانات

وهذه الوثائق الأربعة معاً في مجملها تمثل بشكل دقيق نموذج النظام الذي تجري دراسته.

يشرح الشكل (1.9) عملية النمذجة حيث يبدأ المحلل بتجميع المعلومات لإعداد النموذج المادي للنظام الحالي. ثم يقوم بتبسيط هذا النموذج للتركيز فقط على العمليات الجوهرية فيه، وتسمى عملية التبسيط هذه تحويل النموذج المادي إلى النموذج المنطقي للنظام الحالي، ويخضع المحلل هذا النموذج المنطقي للبحث والدراسة لإجراء التعديلات والإضافات اللازمة فيه في ضوء المشكلات الموجودة في النظام الحالي، والمتطلبات أو الاحتياجات الجديدة للمستخدم. وفي النهاية يتم التوصل إلى



شكل (1.9) خطوات نمذجة الأنظمة

نموذج جديد يسمى النموذج المنطقي للنظام الجديد، ويستخدم هذا النموذج كأساس لتصميم النظام الجديد، من خلال تحديد الطريقة المثلى لتجسيد النموذج المنطقي للنظام الجديد بشكل مادي، وبذلك يتم التوصل إلى النموذج المادي للنظام الجديد.

ونظراً لان عملية النمذجة هذه تبدأ مع بداية مرحلة تحليل الأنظمة وتستمر خلال مرحلة التصميم العام (أو التصميم الأولي) فإننا سنقدم شرحاً كاملاً للخطوتين الأولى والثانية في هذه العملية، لكونهما تتمان خلال تحليل النظام، أما الخطوات الثالثة والرابعة فسنقوم بدراستها في الفصل التالي عند دراسة مرحلة التصميم. ونظراً لأهمية نماذج النظام الحالي ودورها في تحليل النظام الحالي والتعرف على عملياته ومشاكله ونقاط الضعف فيه، وبالتالي تحديد أهداف ومتطلبات النظام الجديد فسنقوم في نهاية هذا الفصل بدراسة كيفية تحليل مشكلات النظام الحالي والبحث عن الحلول المناسبة لها وصياغة أهداف النظام الجديد.

3. خطوات نمذجة الأنظمة:

1.3 بناء النموذج المادي للنظام الحالي.

يتم بناء النموذج المادي للنظام الحالي Current Physical Model خلال مرحلة التحليل، في ضوء المعلومات والحقائق التي يقوم المحلل بتجميعها أولاً بأول وفق منهجية التحليل من الأعلى للأسفل Top-Down Analysis. ففي البداية يرسم المحلل المخطط البيئي للنظام الحالي في ضوء المعلومات التي يتم تجميعها من المقابلات الشخصية مع الإدارة ومع المسؤولين الرئيسيين عن النظام. ويراجع هذا المخطط معهم للتأكد من صحته. بعد ذلك، وبالتعاون أيضا مع المستخدمين الرئيسيين للنظام يقوم بتفكيك النظام الذي تجري دراسته إلى مكوناته الوظيفية Functional Units

ويحدد العلاقات بين هذه المكونات، ويرسم في ضوء ذلك المخطط العام لتدفق البيانات في النظام، وكذلك يقوم بتدقيق هذا المخطط مع أعضاء فريق التطوير للتأكد من صحة ودقة تمثيله للنظام الذي تجري دراسته.

بعد ذلك ينتقل المحلل إلى دراسة كل مكون من المكونات الوظيفية للنظام على حدة، ويرسم المخططات التفصيلية لتدفق البيانات فيها، وقد تمتد هذه المخططات التفصيلية كما ذكرنا سابقاً إلى عدة مستويات أدنى، وبعد الانتهاء من المخططات التفصيلية لكل مكون من المكونات الوظيفية الرئيسة للنظام، ينتقل إلى المكون التالي وهكذا.

تكون المخططات الأولى التي يرسمها المحلل عادة مخططات مادية أي تشرح كيفية أداء عمليات النظام، ويقوم المحلل بإعدادها لاستخدامها كوسيلة لفهم النظام وكأداة لتحليله وتحديد وظائفه الأساسية وعملياته ومعرفة العمليات غير الهامة أو غير الضرورية التي يمكن الاستغناء عنها في النظام الجديد، وكذلك كأداة لمعرفة نقاط الضعف والقصور التي تسبب المشاكل الموجودة في النظام تمهيداً لمعالجتها وإيجاد الحلول المناسبة بشأنها.

يتكون النظام المادي للنظام الحالي عادة من الأدوات الأربعة التي سبق ذكرها وهي : مخططات تدفق البيانات ومخططات الكينونة -العلاقة وتوصيف العمليات وقاموس البيانات. ويجب البدء بإعداد هذه الوثائق منذ بداية أنشطة التحليل ويستمر تطويرها وإضافة المعلومات إليها بشكل تدريجي خلال مراحل التحليل والتصميم.

ويعتبر النموذج المادي للنظام الحالي مرجعاً ضرورياً يمكن الاستفادة منه خلال دورة حياة التطوير، كما يتطلب الأمر معرفة كيفية سير العمل في النظام

الحالي، ولكي يكون هذا النموذج واضحاً وسهل الفهم يجب أن يكون بسيطاً أي ألا يتم إغراقه بالتفاصيل الدقيقة التي تكون غالباً غير هامة لعملية التحليل، وعندما نكون بصدد تصميم نظام جديد تماماً، أي لا يكون لدينا نموذج مادي للنظام الحالي يمكن البدء مباشرة بإعداد النموذج المنطقي للنظام الجديد.

وأخيراً وبعد الانتهاء من إعداد النموذج المادي للنظام الحالي يقوم المحلل بمراجعته وتدقيقه مع الإدارة ومع المستخدمين بهدف الكشف عن الأخطاء التي يمكن أن تكون موجودة فيه، أو العناصر التي يمكن أن يكون قد نسيها المحلل أو غفل عنها، وتعتبر مهمة من وجهة نظر الإدارة أو المستخدمين. كما يمكن خلال هذه المراجعة أن يتم حذف بعض العمليات أو الوظائف ومن الواضح أن هذه المراجعة الشاملة تهدف إلى الوصول إلى نموذج صحيح للنظام الذي تجري دراسته بحيث تعكس صورة حقيقية لواقع العمل في النظام الحالي.

2.3 إعداد النموذج المنطقي للنظام الحالي:

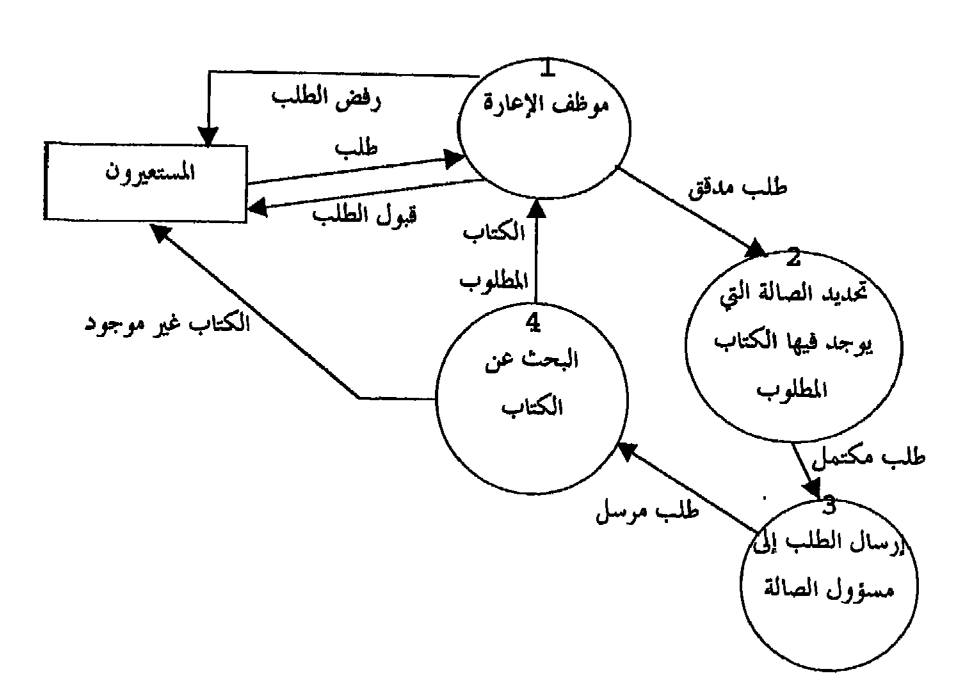
لقد ذكرنا سابقاً أن النموذج المنطقي يهتم بشكل رئيسي بالعمليات والوظائف الجوهرية أو الأساسية في النظام، مهملاً جميع التفاصيل المتعلقة بكيفية تجسيد هذه الوظائف والعمليات بشكل فعلي على ارض الواقع. ولذلك فانه يتم إعداد النموذج المنطقي للنظام الحالي Current Logical System Model من خلال حذف جميع الوظائف والعمليات الغير جوهرية، وكذلك جميع التفاصيل المتعلقة بطرق التنفيذ، من النموذج المادي للنظام الحالي الذي تم إعداده في الخطوة السابقة، على النحو المبين في الشكل (1.9). ويقصد بالعمليات غير الجوهرية تلك العمليات التي لا يتم خلالها إجراء أي تحويل Transformation للبيانات من شكل إلى آخر، أو

العمليات المساعدة المرتبطة بعمليات جوهرية كعمليات تشكيل السجلات وكتابتها في الملف وغيرها.

لقد ذكرنا سابقاً أن النموذج المادي للنظام يتكون من عدد من الوثائق والمخططات. ونظراً لان مخططات تدفق البيانات تمثل عمليات النظام، فان هذه المخططات هي التي تتضمن عادة الصورة المادية للنظام فهي تحوي جميع المعلومات المتعلقة بكيفية تنفيذ وظائفه وعملياته، ولذلك فان التحول من النموذج المادي للنظام إلى النموذج المنطقي يقصد به بصورة رئيسية تحويل المخططات المادية لتدفق بيانات النظام Physical DFD إلى مخططات منطقية Logical DFD، وتتم عملية التحويل هذه من خلال الخطوات التالية:

أ) **حذف جميع العمليات غير الجوهرية** الموجودة في المخطط المادي، ونذكر ثانية بان العمليات غير الجوهرية هي تلك العمليات التي لا يتم خلالها إجراء أي تحويل للبيانات. وفي المخطط المادي المبين في الشكل (2.9) نرى أن العمليات الأربعة في ذلك المخطط هي عمليات مادية حيث:

- العملية 1 موظف الإعارة يستلم الطلبات من المستعيرين ويتأكد من بطاقات الإعارة الخاصة بهم، بأنه يحق لهم استعارة كتب أخرى، فإما يرفض الطلب أو يقبله، ليصبح الطلب مدققا وينتقل إلى العملية المادية التالية:
- العملية 2 تحديد الصالة التي يوجد بها الكتاب المطلوب، حيث أن المكتبة كبيرة وتمتد في عدة طوابق وكل طابق مقسم إلى صالات يتم في كل منها الاحتفاظ بالكتب العائدة لموضوع محدد.



شكل (2.9) المخطط المادي لتدفق البيانات في نظام معالجة طلبات الاستعارة في المكتبة

- العملية 3 : إرسال الطلب إلى الصالة التي يوجد فيها الكتاب المطلوب لإحضاره.
- العملية 4 ويتم فيها البحث عن الكتاب في الرفوف، فإذا كان موجوداً يتم سحبه وإرساله إلى موظف الإعارة أما إذا لم يكن موجوداً فيتم إبلاغ صاحب الطلب بذلك.

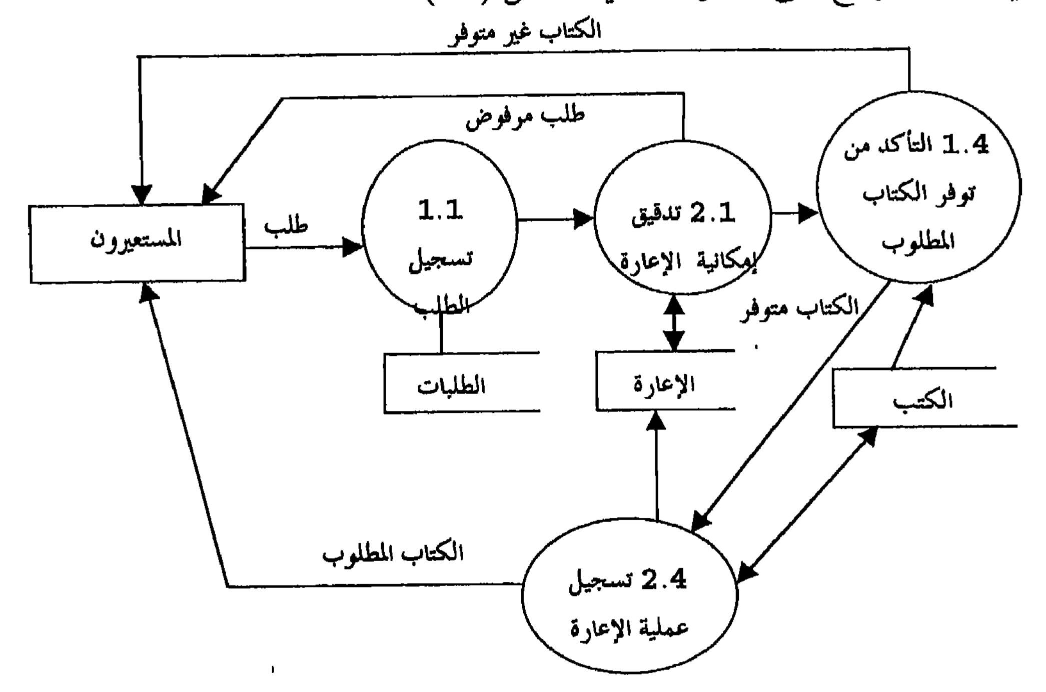
عند دراسة العمليات المبينة أعلاه نجد أن عمليتان منهما غير جوهريتان حيث لا يتم خلالهما أي تحويل للبيانات، وهذه العمليات هي: "تحديد الصالة التي يوجد فيها الكتاب" و " إرسال الكتاب إلى الصالة" ولذلك يمكن حذفهما من

المخطط أما العمليات الباقية في المخطط فتتضمن إجراء تحويل للبيانات وتبقى في المخطط.

ب) استبدال العمليات المادية المتبقية بمكافئاها المنطقية:

حيث يمكن التوسع في العمليات التي يتم خلالها تحويل البيانات (العمليات التي لم يتم حذفها من المخطط) لتعكس الوظائف أو العمليات المنطقية التي تتضمنها عملية التحويل هذه.

ويتم ذلك بتحديد الوظائف المنطقية التي تتضمنها كل عملية ورسمها في المخطط الجديد الموسع بدلاً من العملية التي تمثلها، وهكذا نقوم برسم الوظائف (العمليات) المنطقية للعمليتين الماديتين اللتين بقيتا في المخطط المادي (الشكل 2.9) في مخطط موسع على النحو المبين في الشكل (3.9).

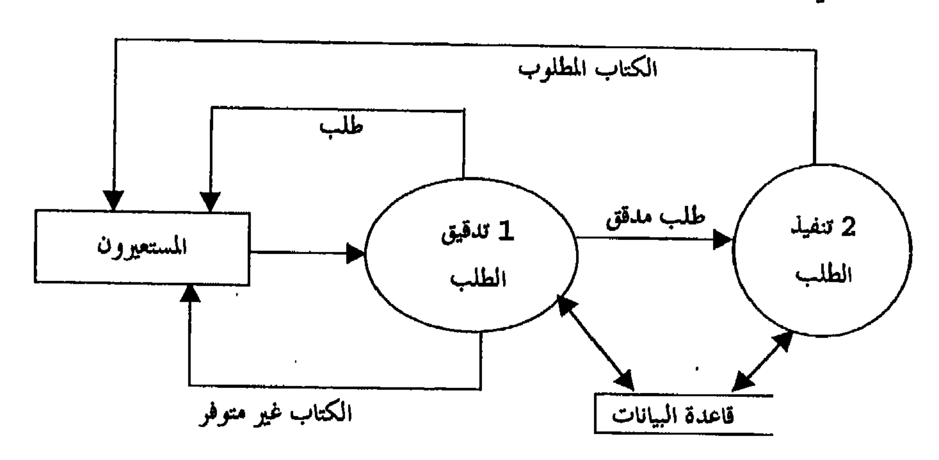


شكل (3.9) المخطط المنطقي لتدفق البيانات في نظام الإعارة

نلاحظ في الشكل (3.9) أنه تم استبدال العملية المادية المسماة " " موظـــف الإعارة" بعمليتين منطقيتين هما : " تسجيل الطلب" و " تدقيق إمكانية الإعـــارة"، وكذلك استبدال العملية " البحث عن الكتاب" بعمليتين منطقيتين هما: " التأكد من توفر الكتاب المطلوب" و " تسجيل عملية الإعارة"، كما نلاحظ تعديل تســـميات تدفقات البيانات لتعكس التفاعلات اللازمة بين هذه العمليات، ونلاحـــظ أيضـــا ظهور مخازن البيانات في هذا المخطط المنطقي الموسع.

جـــ) **تجميع العمليات الموجودة في المخطط الموسع** ورسم مخطط المستوى الأعلـــى له.

في نهاية هذه الخطوة نحصل على المخطط المنطقي لتدفق البيانـــات الـــذي يكافئ المخطط المادي المبين في الشكل (2.9)، ويتم ذلك من خلال تجميع العمليات المنطقية التي تم التوصل إليها في الخطوة السابقة في مجموعات متجانسة ورسمـــها في مخطط منطقي في المستوى الأعلى كما هو مبين في الشكل (4.9) التالي:



شكل (4.9) المخطط المنطقي لتدفق البيانات في نظام الإعارة (المستوى الأعلى)

وهكذا نلاحظ انه تم تجميع العمليات " تسجيل الطلب" و " تدقيق الطلب" و " التأكد من توفر الكتاب" في عملية واحدة هي " تدقيق الطلب" كما تم استبدال تسمية عملية " تسجيل الإعارة" بـــ " تنفيذ الطلب".

3.3 إعداد النموذج المنطقي للنظام الجديد

لقد ذكرنا سابقا أن عملية النمذجة تمتد خلال مرحلتي تحليـــل وتصميـــم النظام، ففي مرحلة التحليل يتم إعداد المخططات الماديـــة والمخططـــات المنطقيـــة للنظـــام الحالي اللذين يستخدمان في فهم النظام والتعرف على مشـــاكله ونقـــاط الضعف والقصور فيه، وبالتالي تحديد أهداف النظام الجديد المطلوب تطويره. أما في مرحلة التصميم فيتم إعداد النموذج المنطقي للنظام الجديد الذي يكون عبارة عـــن تعديل النموذج الحالي في ضوء الاحتياجات الجديدة المنبثقة عن أهداف النظام، كما هو مبين في الشكل (1.9) ، وسندرس هذه الخطوة في الجزء الثالث من هذا الكتاب في فصل " تصميم النظام الجديد" وهي تسمى مرحلة التصميم العام للنظـام Broad System Design.

4.3 إعداد النموذج المادي للنظام الجديد

وهي الخطوة الأخيرة في عملية النمذجة وتتم أيضا في مرحلـــة التصميـــم ويجري خلالها تحديد المواصفات التصميمية التفصيلية لتنفيذ النظام الجديد. ونكتفـــي بهذا القدر من الشرح الآن لأنه سيتم دراستها بشكل تفصيلي في الفصل القـــادم " تصميم النظام الجديد"

4. أهداف النظام

يهدف تحليل الأنظمة وتصميمها إلى تطوير أنظمة معلومات جديدة تلبي متطلبات المستخدم، التي يتم صياغتها عادة بشكل مجموعة من الأهداف، يطلق عليها تسمية أهداف النظام System Objectives ، وتمثل الهدف أو الغاية التي تتوجه نحوها جميع الأنشطة التي تتضمنها مرحلة التصميم. ولذلك تهتم مرحلة التحليل بدراسة مشكلات النظام الحالي وإيجاد الحلول المناسبة لها، أي صياغة الأهداف التي يجب أن يحققها النظام الجديد. ونظراً لأهمية التحديد الصحيح والصياغة الواضحة والدقيقة لهذه الأهداف وما لها من تأثير مباشر في جودة النظام الذي سيتم تطويره فإننا سندرسها في هذه الفقرة.

أن أهداف النظام، كما ذكرنا أعلاه ، تعتبر الموجه الرئيسي لمرحلة التصميم، حيث تهدف جميع أنشطة التصميم التي سندرسها في الفصول القادمة إلى تلبية هذه الأهداف بأقصى كفاءة وفاعلية ممكنة، ويمكن أن تتضمن أهداف النظام التي يتم صياغتها خلال مرحلة التحليل في ضوء مشاكل النظام الحالي ومتطلبات تحسينه ما يلي: تحسين العمليات التي يقوم بها النظام من حيث زيادة إنتاجيتها أو كفاءتها أو فاعليتها وتحديد الطريقة التي سيتم فيها قياس حجم هذه التحسينات.

ولذلك يجب أن تكون الأهداف ذات طبيعة قابلة للقياس الكمي فمثلاً الهدف التالي " يجب أن لا تزيد نسبة الأخطاء في عمليات الإدخال عن 1% تعتبر صياغته واضحة ودقيقه بالمقارنة مع الصياغة التالية " يجب تقليل الأخطاء إلى أدنى حد ممكن".

1.2 أنواع الأهداف

يمكن أن يهدف النظام الجديد إلى تحقيق واحد أو اكثر من مجموعات الأهداف التالية:

أ) أهداف وظيفية Functional Objectives:

ويقصد بها إدخال وظائف أو عمليات جديدة إلى النظام الحالي لتلبية احتياجات المستخدمين، وتكون هذه العمليات بالطبع غير موجودة في النظام الحالي، ولكن التغيرات الجارية في بيئة العمل قد أدت إلى ظهور احتياجات جديدة يجب تلبيتها في النظام الجديد.

كما يمكن أن تتضمن هذه الأهداف إدخال تعديلات في وظائف موجودة، مثلاً تعديل وظائف الحماية في النظام ليصبح اختراقها اصعب مما هو عليه الأمر حالياً.

ب) أهداف تتعلق بتحسين العمليات Process Improvements Objectives

تتعلق هذه الأهداف غالباً بإعادة تصميم عمليات النظام الحالي بهدف تحسينها، فمثلاً تغير طريقة الوصول إلى البيانات، باستبدال الوصول المتسلسل Sequential Access بالوصول المباشر Direct Access أو تغيير تسلسل تنفيذ عمليات معينة ليجري بأسلوب اكثر كفاءة وهكذا.

جـ) أهداف عملياته Operational Objectives

تتعلق هذه الأهداف بتحديد معايير الأداء التي يجب أن يحققها النظام الجيد، وتتعلق غالباً هذه المعايير بالدقة Accuracy (العمليات والمخرجات) وسرعة

الاستجابة، وحجم العمليات التي يمكن للنظام تنفيذها خلال فترة زمنية معينة وهكذا.

د) أهداف تتعلق بتحسين بيئة عمل الأفراد الذين يستخدمون النظام وزيادة رضاهم الوظيفي.

تتضمن هذه المجموعة أهدافا في غاية الأهمية، يتم من خلالها توفير المتطلبات الضرورية لنجاح النظام الجديد وقبوله، وبالتالي استخدامه من قبل الأفراد والإدارات في المنظمة، تتعلق هذه الأهداف غالباً بسهولة الاستخدام، وبما يوفره النظام من إمكانات وتسهيلات تجعل عمل الموظفين اكثر تحدياً، وتبعد كل ما يمكن أن يجعله مملاً وروتينياً، ويسمى هذا التوجه إغناء العمل Job Enrichments

وحتى لا يكون عدد أهداف النظام كبيراً فانه يتم أيضا صياغتها بشكل هرمي من الأعلى إلى الأسفل، أي وضعها ضمن بنية هرمية Hierarchical Structure، تبدأ من الأعلى بالأهداف العامة (العليا) High-level Objectives، ثم يتم تفصيلها إلى المستويات الأدنى بشكل أهداف تفصيلية يمكن أن تقع في عدة مستويات. كما يجب أيضا عند تحديد الأهداف التمييز فيما بينها من حيث درجة أهميتها واختيار الأهداف الرئيسة Key Objectives وترتيبها وفق أولوياتها، ويتم تحديد هذه الأولويات بالتعاون مع الإدارة والمستخدمين. وبعد ذلك لا بد من تحديد قيم كمية مستهدفة لهذه الأهداف الرئيسة، فإذا ما تم التوصل فعلاً إلى هذه القيم يمكن اعتبار النظام قد حقق أهدافه فعلاً، والعكس صحيح أيضا.

2.2 تحديد أهداف النظام

يبدأ تحديد أهداف النظام الجديد مع بداية مرحلة التحليل، ويستمر خلال عمليات التحليل، وينتهي مع نهاية هذه المرحلة فعندما يقوم المحلل بتجميع المعلومات ومقابلة المستخدمين ودراسة الوثائق والبرامج الحاسوبية المستخدمة في النظام الحالي يبدأ باكتشاف نقاط الضعف في هذا النظام والوقوف على أسباب قصوره عن تلبية الاحتياجات الحالية والمستقبلية للمنظمة، وبشكل عام يتم تحديد أهداف النظام في مرحلة التحليل وفق الخطوات التالية:

1- **تحديد مشكلات النظام الحالي** وأسباب قصوره عن تلبية استراتيجيات العمل في المنظمة، ويتم ذلك من خلال تجميع المعلومات عن استراتيجيات المنظمة الحالية والمستقبلية وعن الأداء الفعلي للنظام الحالي والاحتياجات المختلفة للمستخدمين، وكذلك المشكلات والعوامل المختلفة التي تؤثر على أداءه الحالي. ومن خلال فهم طريقة عمل النظام الحالي (النموذج المادي للنظام الحالي) ودراسة العوامل المذكورة أعلاه يحدد المحلل قائمة بنقاط الضعف أو أوجه القصور الموجودة في هذا النظام (مشكلات النظام الحالي) .

2. **صياغة أهداف النظام الجديد** في ضوء القائمة التي تم التوصل اليها في الخطوة السابقة. فمثلاً إذا كان هناك مشكله تتعلق بعدم توفر جميع المعلومات اللازمة لتنفيذ عمل معين أو اتخاذ قرار ما، فيكون الهدف المقابل لها هو " يجب أن يوفر النظام الجديد المعلومات التالية ...."، أما إذا كانت المشكلة هي بطئ سير العمليات في النظام الحالي مما يجعل الوثائق تتراكم

ويؤخر عمليات المعالجة، فان الهدف المقابل يمكن أن يكون مثلاً وأخيراً يجب أن يتمكن النظام من إدخال 100 فاتورة في الساعة "إذا كانت المشكلة أن النظام الحالي لا يوفر معلومات تلخيصية عن المبيعات فيمكن صياغة الهادف التالي: توفير إمكانية الحصول على تقارير تلخيصية عن حركة المبيعات" وهكذا.

ويجب أن لا ننسى أيضا ضرورة مراجعة وتدقيق الأهداف بعد الانتهاء من صياغتها مع كل من الإدارة والمستفيدين لإدخال التعديلات اللازمة فيها.

3- مخرجات مرحلة التحليل.

في نهاية مرحلة التحليل يقوم المحلل بإعداد ملف أو تقرير يسمى مواصفات النظام System Specification يتضمن ما يلي:

1- وصف مختصر لمشاكل النظام الحالي.

2- أهداف النظام الجديد المطلوب.

3- القيود المتعلقة بالموارد المتاحة لتطوير النظام.

4- مقاييس أو معايير قبول النظام.

5- نموذج النظام الحالي وهذا يشمل:

- مخططات تدفق البيانات (DFDs)
- مخططات الكينونة – العلاقة (E-RD)
- توصيف العمليات (PD)
- قاموس بيانات النظام (DD)

ويطلق على هذا الملف أيضا توصيف المشـــكلة "Problem Specification" وتكون محتوياته أساس العمل في المرحلة التالية وهي مرحلة التصميم، ولذلك يجـــب أن تكون وثائقه دقيقة وواضحة ومكتملة.

أسئلة الفصل

1- ما هي النماذج ولماذا تستخدم؟ وما هو المقصود بعملية النمذجة؟

2- اشرح خطوات نمذجة الأنظمة؟

3- اشرح الخطوات المتبعة للتحول من النموذج المادي إلى النموذج المنطقي للنظـــام الحالي؟

4- ما أهمية تحديد أهداف النظام الجديد وكيف يمكن تنفيذ هذه الأهداف؟

5- اشرح الخطوات المتبعة لتحديد أهداف النظام الجديد؟

6- ما هو المقصود بمواصفات النظام وما هي الوثائق التي يمكن أن تتضمنها؟



تصميم أنظمة المعلومات الحاسوبية



الفصل العاشر

تصميم النظام الجديد

تهتم مرحلة التصميم بالوصول إلى الحلول التصميمية المثلى لبنـــاء النظـــام الجديد. ولذلك فهي تحتاج إلى مهارات إبداعية خلاقة، لكونها تتعامل مـــع نظـــام جديد ما يزال غير موجوداً سوى في مخيلة وأفكار المجموعة التي تقـــوم بـــالتصميم. وذلك بعكس مرحلة التحليل السابقة والتي يجري التعامل فيها مع النظـــام الحـــالي وهـــو نظام واقعي موجود فعلاً، تتم دراسته ونمذجته وتحديد مشكلاته ومتطلبـــات حلها. فإذا تمت عملية التحليل بطريقة صحيحة يكون نموذج النظام الحالي النـــاتج عنها نموذجاً صحيحاً. أما في مرحلة التصميم فإننا لا نستطيع القول بان تصميم مـــا صحيحاً وآخر غير صحيح، فالعملية ذات طبيعة إبداعية يمكن الوصول من خلالهـــا إلى تصاميم متعددة تتفاوت إلى حد كبير فيما بينها. ولذلك نتحدث في هذه المرحلة عن التصميم الأفضل أو الأمثل، أي الذي يلبي متطلبات النظام بطريقة اكثر كفـــاءه وفاعلية. إن عملية التصميم في جوهرها هي عملية حل مشـــكلات ، أي يجـــري البحث خلالها عن افضل الحلول التصميمية لبنـــاء نظـــم ذات أهـــداف محـــدده. فالأهداف هي المعيار الأول والأساسي الذي يتم على أساسه تقييم جودة التصـــاميم التي يتم التوصل إليها. ولذلك تنطلق مرحلة تصميم النظام مـــن أهـــداف النظـــام المطلوب تصميمه، والتي تم تحديدها في مرحلة التحليل السابقة. لقد أشـــرنا عنـــد دراسة دورة حياة تطوير النظام إلى أن مرحلة التصميم تتم في مرحلتين فرعيتين همـــا **التصميم العام للنظام**، والتي يجري البحث فيها عن افضل الحلول التصميميه لبنـــاء

النظام الجديد، و**التصميم التفصيلي للنظام** حيث توضع التصاميم التفصيلية للحل الذي تم التوصل إليه في الخطوة السابقة. وسنتعرف في هذا الفصل على أهم الأنشطة التي تتكون منها هاتين المرحلتين الفرعيتين ولكننا سنركز على دراسة المرحلة الأولى فقط (التصميم العام للنظام) نظراً لأن أنشطة التصميم التفصيلي سندرسها في الفصول الثلاث القادمة.

1- التصميم العام للنظام: Broad System Design

وتسمى أيضا هذه المرحلة بالتصميم الأولي للنظام Preliminary System Design. تبدأ هذه المرحلة فور الإنتهاء من أعداد النموذج المنطقي للنظام الحالي، وكما هو مبين في الشكل (1.9) في الفصل السابق، يتم أعداد النموذج المنطقي للنظام الجديد اعتمادا على النموذج المنطقي للنظام الحالي وأهداف النظام الجديد الذين تم تحديدهما خلال مرحلة التحليل.

وهكذا فان الهدف الرئيس لهذه المرحلة هو بناء نموذج النظام الجديد الذي يلبي الأهداف التي تم تحديدها. أي أنه خلال هذه المرحلة يتم التحول من النموذج المنطقي للنظام الحالي إلى النموذج المادي للنظام الجديد. و تتطلب هذه المرحلة من المحلل **مهارات إبداعية وقدرة على التحليل والابتكار والتجديد** فتحقيق أهداف النظام يمكن أن تتم بطرق متعددة، وعلى مصمم النظام أن يبحث كل البدائل التي يمكن من خلالها تحقيق هذه الأهداف واختيار الطريقة المثلى لذلك، واعتمادها لإعداد النموذج المنطقي للنظام الجديد.

أما الأنشطة التي يجب أن يتبعها المحلل خلال هذه المرحلة فهي:

1.1 تصميم النموذج المنطقي للنظام الجديد

يتم تصميم النموذج المنطقي للنظام الجديد انطلاقا من النموذج المنطقي للنظام الحالي، وذلك بإضافة الوظائف الجديدة التي يرغب بها المستخدم (أهداف النظام الجديد). وهنا ندخل للمرة الأولى في مجال التصميم والذي يتم خلاله تقرير ماذا يجب عمله فعلاً ، وتحديد الطريقة المنطقية للقيام بذلك.

إذا تطلبت الإحتياجات الجديدة إجراء تعديلات في الأنشطة (العمليات) الحالية، فإننا نقوم بتغيير هذه العمليات في النموذج المنطقي للنظام الحالي، وكذلك في مخططات تدفق البيانات في المستويات الأدنى في المخطط العام. أما إذا كانت الوظيفة المطلوبة عبارة عن نشاط جديد تماماً، فإننا نرسم العمليات المتعلقة بهذه الأنشطة الجديدة في مخططات تدفق البيانات ذات المستوى الأدنى، ونضيف كذلك هذه العملية إلى المخطط العام للنظام . وفي جميع الأحوال يجب بالطبع تعديل مخطط هيكل البيانات وقاموس البيانات لتعكس الكينونات Entities وعناصر البيانات الجديدة.

فمثلاً لنفرض أن المستخدم يطلب أن يوفر له النظام الجديد القدرة على الوصول إلى شاشة تظهر بيانات الفواتير المتأخرة لمورّد معين، وهذه الشاشة يجب أن تتضمن رمز المورد واسمه وأرقام الفواتير وقيمها وتاريخ استحقاق دفعها، وهذه العملية غير موجودة في النظام الحالي. لذلك يتم إضافة هذه العملية إلى مخطط تدفق البيانات للنظام الجديد، وربطها مع مخازن البيانات اللازمة لها وتحديد التدفقات الداخلة إليها والخارجة منها.

وبنفس الطريقة يمكن إجراء جميع الإضافات والتغيرات المطلوبة في النظام الجديد، في مخططات التدفق. ويجب أن لا ننسى أيضا إجراء التعديلات في

المستويات الأخرى لمخططات تدفق البيانات، وكذلك تعديل مخططات الكينونـــة- العلاقة بما ينسجم مع هذه التعديلات، وأخيرا يتم إضافة هذه التغييرات إلى قـــاموس البيانات وتوصيف العمليات الجديدة.

ونظراً لأهمية عملية تصميم النموذج المنطقي للنظام الجديد سنتعرف بشكل تفصيلي إلى خطوات تنفيذها المبينة في الشكل (1.10) وهي :

1) فحص العمليات الموجودة في النموذج المنطقـــي للنظـــام الحـــالي و**تحديـــد العمليات** التي تتطلب أهداف النظام الجديد تغييرها أو تعديلها، وهكذا يتـــم تحديد تأثير أهداف النظام الجديد على العمليات الموجودة في النظام الحالي.

2) **تحديد منطقة التغييرات Domain of Change** في النموذج المنطقي الحـــالي: والتي تشمل جميع العمليات التي تم تحديدها في الخطوة الســـابقة باعتبارهـــا ستتأثر بأهداف النظام الجديد. كما تشمل منطقة التغيـــيرات هـــذه تلـــك العمليات التي تمثل واجهة تفاعل بين هذه العمليات. ويمكن أن تتكون هـــذه المنطقة من عمليات مرتبطة معاً أو عمليات مستقلة، أو يمكن أن تشمل جميع عمليات النظام، وهكذا يتم في هذه الخطوة تحديد الأجزاء أو المناطق الـــتي سيطالها التغيير في النظام الجديد الذي يجري تصميمه.

3) **تحديد طريقة التفاعل** بين المنطقة أو المناطق التي سيطالها التغيير في النمـــوذج (المخطط) وتلك التي ستبقى دون تغيير. وهذا يعـــني تصميـــم واجـــهات Interface التعامل بين العمليات التي سيطالها التغيير في النظام الجديد وبـــين بقية عمليات النظام .

النموذج المنطقي للنظام الحالي

أهداف النظام

1
تحديد العمليات التي سيطالها التغيير

العمليات التي يجب تغييرها

2
تحديد منطقة التغييرات

منطقة التغيرات

3
تصميم واجهة التفاعل بين مناطق التغير وبقية أجزاء النظام

4
إعداد المخطط المنطقي للنظام الجديد

تعديل العمليات في مناطق التغيير

إعادة تصميم مناطق التغير

النموذج المنطقي للنظام الجديد

النموذج المنطقي للنظام الجديد

تحديد مناطق الحوسبة

توليد حلول بديلة للتصميم المادي

حلول تصميمية بديلة

اختيار التصميم الأمثل

اعتبارات فنية واجتماعية واقتصادية

التصميم المادي الأمثل

إعداد النموذج المادي للنظام الجديد

النموذج المادي للنظام الجديد

شكل (1.10) خطوات إعداد النموذج المنطقي ثم المادي للنظام الجديد

4) **إعداد النموذج المنطقي للنظام الجديد** : وذلك بإعادة تصميم العمليات الموجودة في المناطق التي سيطالها التغيير والتي تم تحديدها في الخطوات السابقة، أو الاكتفاء بإجراء التعديلات الضرورية في هذه العمليات.فعندما تتطلب أهداف النظام الجديد إحداث تغييرات كبيرة في العمليات الموجودة في منطقة التغييرات، يتم اللجوء إلى إعادة تصميم هذه العمليات وفق ما تتطلبه الأهداف . أما إذا كانت التغييرات المطلوبة بسيطة ومحدودة فيمكن الاكتفاء بإجراء تعديلات في العمليات الحالية على النحو الذي يلبي أهداف النظام الجديد .وسنشرح فيما يلي إجراءات كل من الطريقتين أعلاه :

الطريقة الأولى : إعادة تصميم العمليات :

يتم إعادة تصميم المناطق التي سيشملها التغيير في مخططات التدفق وفق الخطوات التالية والمبينة في الشكل (2.10) :

أ) رسم شكل دائري كبير وفارغ لتمثيل منطقة التغييرات التي سيتم إعادة تصميم عملياتها.

ب) تحديد مخازن البيانات اللازمة داخل هذه المنطقة، ورسمها ضمن الشكل الدائري المذكور أعلاه.

ج) تتبع كل تدفق من التدفقات الداخلة إلى منطقة التغييرات وتحديد العمليات التي يجب أن تتم عليه حتى يستقر في أحد مخازن البيانات،أو يخرج إلى خارج منطقة التغييرات.ورسم هذه التدفقات والعمليات ضمن الشكل الدائري الذي يمثل المنطقة التي يجري إعادة تصميمها .

د) تحديد العمليات اللازمة لتحويل البيانات ضمن منطقة التغييرات ، أي تلك العمليات التي تستخدم البيانات الموجودة في مخازن البيانات وتتبع مخرجاتها . ورسم

هذه العمليات وتدفقاتها ضمن الشكل الدائري الذي يمثل المنطقة التي يجري إعـــادة تصميمها (منطقة التغييرات).

هـــ) تحديد العمليات التي تقوم بإعداد المخرجات أي التدفقات الخارجة من منطقة التغييرات ، ورسم هذه العمليات وتدفقاتها في منطقة التغييرات .

الطريقة الثانية : إجراء تعديلات في منطقة التغييرات:

يمكن اللجوء إلى هذه الطريقة عندما يكون حجم التغييرات المطلوبة بسيطاً ، فبـدلاً من إعادة تصميم منطقة التغييرات بشكل كامل ، يمكن إجراء بعض التغيـــيرات في العمليات الموجودة فيها . ويمكن أن تشمل هذه التغييرات ما يلي :

- إضافة عمليات وإنشاء تدفقات جديده.
- إضافة مخازن بيانات جديده وتعديل العمليات في ضوء ذلك .
- تغيير تسلسل العمليات .
- حذف العمليات غير الضرورية .
- دمج عملتين أو اكثر .
- تعديل بعض العمليات بإضافة بيانات جديدة إليها .

2.1 تصميم النموذج المادي للنظام الجديد :

وهي الخطوة الأخيرة في عملية النمذجة . فبعد الانتهاء من إعداد النمـــوذج المنطقي للنظام الجديد الذي يلبي أهداف النظام ، يقوم المصمم بـــإعداد النمـــوذج المادي الذي يعكس التجسيد المادي لهذا النظام ، والذي يتضمن جميع التفـــاصيل المتعلقة بمتطلبات تنفيذه بما في ذلك إعداد مواصفات التجهيزات والبرمجيات اللازمة لبناء هذا النظام .

1. رسم حدود المنطقة الـــتي سيتم إعادة تصميمها

2. تحديد مخازن البيانات

3. تتبع التدفقات الداخلة

4. تحديد العمليـــات اللازمـــة لتحويل البيانات .

5. تحديد العمليات الـــتي تقوم بإعداد المخرجات

شكل (2.10) خطوات إعادة تصميم منطقة التغييرات

أما أهم الأنشطة اللازمة لتصميم النموذج المادي للنظام الجديد فهي كما مبين في الشكل (1.10) :

أ) **البحث عن بدائل مختلفة للتصميم المادي :**

تبدأ عملية البحث عن البدائل المختلفة التي يمكن من خلالها إيجاد الحلول التصميمية المثلى لتصميم النظام الجديد استنادا إلى النموذج المنطقي الذي تم إعداده في المرحلة السابقة . ففي هذا النموذج المنطقي يقوم المصمم بتحديد العمليات التي يمكن حوسبتها وإحاطتها بخط متقطع في مخططات تدفق البيانات (المستوى العام أو المستويات التفصيلية) وتسمى هذه المناطق المرشحة للحوسبة مناطق الاتمته Computerization Boundary . وغالباً يتم تحديد اكثر من منطقة لحوسبتها ومن الضروري هنا توليد عدد من الحلول التصميمية البديلة التي يمكن أن تتراوح بين الحوسبة الشاملة لجميع عمليات النظام Total Computerization وخيار حوسبة عدد محدود من العمليات Minimal Computerization ، بما يلبي أهم متطلبات وأهداف النظام الجديد .

إن أهمية هذه الخطوة تأتي من ضرورة أن يقوم المصمم بتوليد أكبر عدد ممكن من الحلول البديلة . فكلما كان عدد البدائل أكبر كلما كانت عملية البحث عن الحل الأمثل أكثر شمولية ودقة . وتجدر الإشارة هنا إلى أن هذه الحلول البديلة يجب أن تكون ذات طبيعة عامه أي لا تتضمن تفاصيل كثيرة بل تقدم حلول مادية عامة Broad physical design alternatives تمثل مستوى الحوسبة الذي سيتم استخدامه لبناء النظام الجديد.

ب) إختيار التصميم المادي الأمثل :

بعد تحديد اكبر عدد من الحلول التصميمية المتعلقة بالتجسيد المادي للنظام الجديد، تتم في هذه الخطوة المفاضلة بين هذه البدائل لاختيار البديل الذي يتضمن التصميم المادي الأمثل (النموذج المادي للنظام الجديد) .

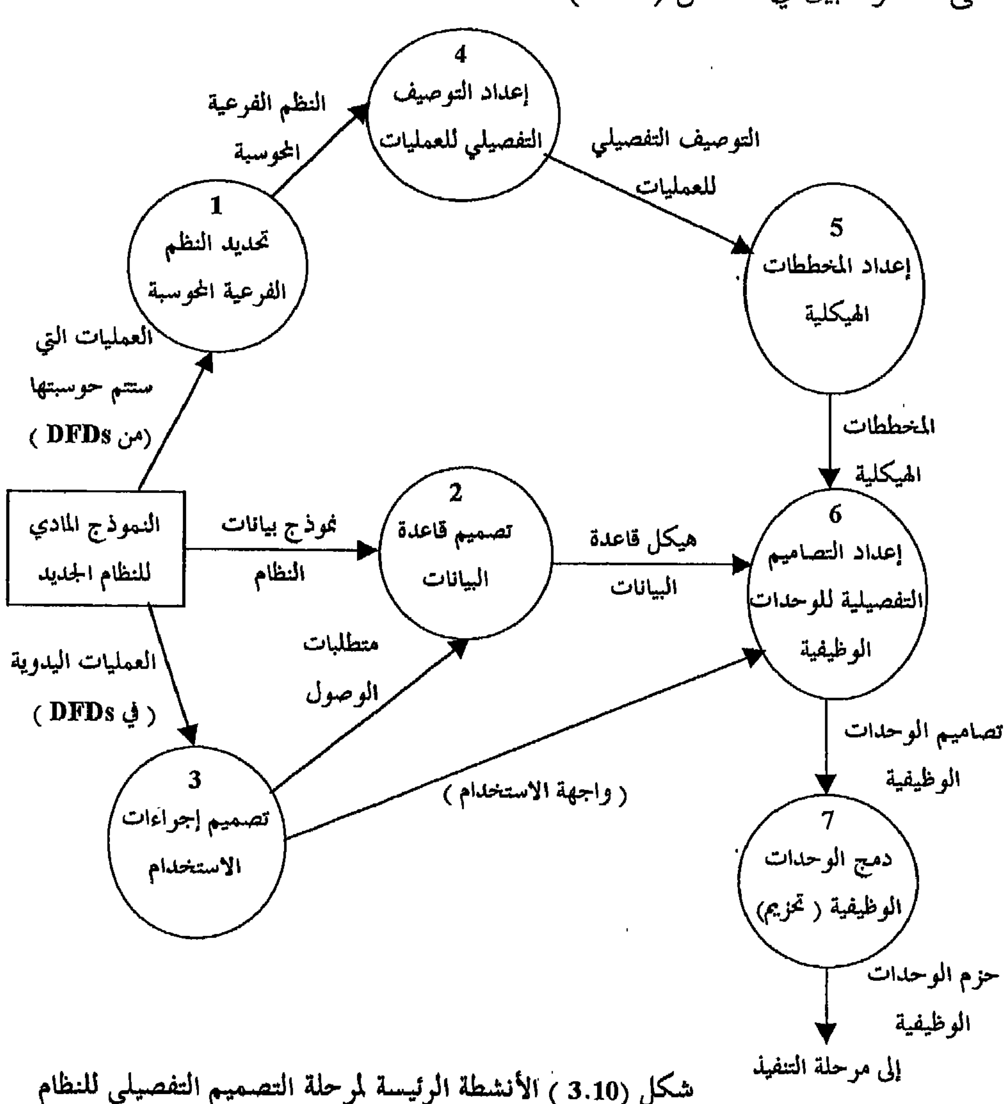
وتتم هذه المفاضلة من خلال دراسة الجدوى الفنية والعملياتية والاقتصادية لهذه البدائل . ولقد تعرفنا في الفصل الخامس من هذا الكتاب إلى طرق دراسة الجدوى ، ولكنها تتم في هذه المرحلة بطريقة أكثر عمقاً وتفصيلاً . وهكذا في نهاية هذه الخطوة يتم اختيار التصميم المادي الأمثل للنظام وإعداد النموذج المادي للنظام الجديد وفقاً لذلك .

وتنتهي هذه الخطوة بمراجعة النموذج المادي للنظام الجديد مع الإدارة والمستخدمين وإدخال التعديلات اللازمة فيه، ثم المصادقة عليه قبل البدء بالمرحلة التالية وهي التصميم التفصيلي للنظام .

2. التصميم التفصيلي للنظام **Detailed System Design:**

يبدأ التصميم التفصيلي للنظام بعد إعداد النموذج المادي للنظام الجديد . حيث يحدد هذا النموذج العمليات ومخازن البيانات التي ستتم حوسبتها وكذلك تلك التي سيستمر تنفيذها بالطريقة اليدوية .

وتتضمن هذه المرحلة إعداد التصاميم التفصيلية اللازمة لتنفيذ نظام المعلومات الجديد وفق النموذج المادي الذي تم التوصل إليه في مرحلة التصميم العام. وتتضمن هذه التصاميم بصورة رئيسية ثلاث عناصر هامة هي : تصميم قاعدة بيانات النظام، وتصميم واجهة الاستخدام، وكذلك تصميم برامج النظام على النحو المبين في الشكل (3.10) .



شكل (3.10) الأنشطة الرئيسة لمرحلة التصميم التفصيلي للنظام

ويتضح من الشكل (3.10) أعلاه أن عمليات التصميم التفصيلي تنطلق من النموذج المادي للنظام الجديد الذي يتم إعداده خلال مرحلة التصميم العام . حيث يستخدم نموذج بيانات النظام (مخطط الكينونة – العلاقة E-RD) أو النموذج العلاقاتي Relational Model) لتصميم النموذج المنطقي ثم المادي لقاعدة بيانات النظام.

أما عمليات النظام (مخططات تدفق البيانات) في النظام الجديد ، والتي تم فيها تحديد " مناطق الحوسبة " خلال مرحلة التصميم العام للنظام ، فتكون العمليات التي ستشملها الحوسبة منطلقا لتصميم برامج النظام ، بينما تستخدم العمليات التي ستبقى يدوية في النظام الجديد أساسا لتصميم إجراءات الاستخدام .

وفي النهاية يتم إعداد التصاميم التفصيلية للوحدات الوظيفية Detailed Module Design استنادا إلى هيكل قاعدة البيانات ومتطلبات الإدخال والإخراج (واجهة الاستخدام) وسندرس في الفصول الثلاث القادمة هذه الأنشطة الرئيسية لمرحلة التصميم التفصيلي للنظام الجديد وذلك على النحو التالي :

- تصميم واجهة استخدام النظام.
- تصميم قاعدة بيانات النظام.
- تصميم برامج النظام.

وبالإضافة إلى الأنشطة الرئيسية لمرحلة التصميم التفصيلي التي سيتم دراستها في الفصول التالية، فإنه من المفيد الإشارة إلى أنشطة التصميم الأخرى والتي تعتبر مهمة أيضا لضمان كفاءة وجودة عمل النظام الجديد وهي :

1. تصميم إجراءات الأمن والحماية في النظام .
2. تصميم شبكة نقل البيانات Data Communication Networks .

3. وضع خطة تنفيذ النظام واختباره .
4. مراجعة التصاميم للتأكد من مطابقتها للأهداف المقررة .

أسئلة الفصل

1. لماذا تقسم مرحلة التصميم إلى مرحلتين فرعيتين هما التصميم العام والتصميم التفصيلي للنظام ؟
2. ما هي خطوات مرحلة التصميم العام للنظام ؟
3. اشرح خطوات تصميم النموذج المنطقي للنظام الجديد ؟
4. اشرح الطرق المختلفة لتصميم النظام المنطقي الجديد ؟
5. ما المقصود بمناطق الحوسبة في مخطط التدفق ؟
6. اشرح الأنشطة الرئيسية لمرحلة التصميم التفصيلي للنظام الجديد ؟
7. عدد أنشطة التصميم الضرورية لضمان كفاءة وجودة عمل النظام الجديد ؟

الفصل الحادي عشر

تصميم واجهات الاستخدام

يقصد بواجهات الاستخدام User Interfaces الطريقة التي يتم من خلالها تفاعل المستخدم مع الحاسوب Man-Computer Interaction . وهذا يعني **النماذج والوثائق** Forms التي يتم استخدامها عند إدخال البيانات إلى الحاسوب ، وكذلك **الشاشات** Screens التي يتم من خلالها التحاور بين المستخدم والحاسوب ، وأيضاً تشمل واجهات الاستخدام **المخرجات المطبوعة** من تقارير وجداول ورسومات بيانية وغيرها.

وتعتبر هذه الواجهات من أهم عناصر النظام الذي يجري تصميمه، فالمستخدمون عادة يحكمون على جودة النظام ، وبالتالي يقبلونه أو يرفضونه من خلال فهمهم لهذه الواجهات وسهولة استخدامها والتدرب عليها . كما أن المستخدمين يقضون معظم ساعات عملهم في تفاعل دائم معها ، وبالتالي فإن رضاهم الوظيفي يرتبط بما توفره لهم هذه الواجهات من بيئة عمل مريحة ، شيقة وسهلة الاستخدام . ونظراً لأهمية هذا الموضوع وضرورة تصميم واجهة استخدام ذات جودة عالية فإننا سندرسها بشكل تفصيلي في هذا الفصل.

1 – خصائص واجهة الاستخدام الجيدة :

يتم الحكم على واجهة الاستخدام من خلال عدد من الخصائص الهامة والتي يأتي في مقدمتها أن تكون **صديقة المستخدم** User-Friendly. وهذا يعني أن **توفر له المساعدة** Helpful وأن تكون مرنة يمكن مواءمتها حسب تفضيلاته Adaptable وأن

تتحمل الأخطاء Tolerant، بالإضافة إلى **الجمالية والراحة** من حيث الأشكال والألوان ، مما يجعل المستخدم لا يمل من التعامل معها ، ويثق باستخدامها. فالرسائل التي تنبه المستخدم مثلاً إلى حدوث خطأ ما في التعامل مع النظام ، دون أن تخبره بالضبط ماذا يجب أن يقوم به لمعالجة هذا الخطأ ، لا يمكن اعتبارها واجهة صديقه للمستخدم. وهكذا فإنه يجب أن يتذكر المصمم دوماً ضرورة تصميم واجهة استخدام صديقة تجعل التعامل بين الإنسان والحاسوب سهلاً وممتعاً واكثر إنتاجية .

كما يجب أن توفر واجهات الاستخدام **بيئة عمل فعالة**، كما هو الحال في بيئة التشغيل Windows التي يمكن من خلالها إظهار جميع المعلومات والأدوات اللازمة لعمل المستخدم على الشاشة التي تسمى سطح المكتب . وهي تمثل فعلاً مكان عمل مناسب جداً للمستخدم .ونظراً لان منطقة العمل هذه يمكن أن تحتوي عدداً كبيراً من العناصر مثل أشرطة القوائم وصناديق الأدوات والنوافذ وغيرها ، فانه من المهم ترتيب وتوزيع هذه الأشياء بشكل مريح ومناسب على شاشة الحاسوب .

وتعتبر قوة التحمل أو الثبات Robustness من الخصائص الهامة أيضاً التي يجب أن تتوفر في واجهات الاستخدام . ويقصد بها ، عدم تعطل النظام عندما يتم استخدامه بطريقة غير صحيحة من قبل المستخدم . وهذا يعني أن أخطاء المستخدم يجب أن لا تؤدي إلى توقف النظام عن العمل. ويتم توفير هذه الإمكانية من خلال تضمين واجهات الاستخدام إجراءات تدقيق تمنع حدوث مثل هذه الأخطاء .

إن واجهة الاستخدام يجب أن تتعامل مع المدخلات غير الصحيحة فتكتشفها أولاً وتمنع دخولها إلى النظام ، وتطلب من المستخدم تصحيحها . وهذا يتم عادة من خلال تصميم **إجراءات الرقابة** Controls اللازمة لاختبار هذه

المدخلات والتأكد من صحتها . كما يجب أن تتميز واجهات الاستخدام **بالكفاءة والفاعلية** Efficiency and Effectiveness حيث يتم من خلال هــــذه الخصـــائص الحكم على ملائمة النظام للاستخدام اليومي الفعلي . تعني الكفاءة قدرة الواجهـــة على تلبية إحتياجات المستخدم بالسرعة المطلوبة ، أما الفاعلية فيقصد بها مطابقـــة الواجهات لتصورات وأفكار ونمط عمل المستخدمين . فمثلا يجب أن تتم عمليـــات الإدخال بأقل جهد ووقت ممكن ، أي باستخدام أقل عدد ممكـــن مـــن ضربـــات المفاتيح، أو باستخدام أساليب الإدخال السريعة ، كالإدخال الضوئي أو من خـــلال رموز الأعمدة Bar Code أو غيرها. وفي حالة استخدام لوحات المفـــاتيح لكتابـــة المدخلات على الشاشة يجب استخدام تسميات واضحة ، والعمل علـــى إدخـــال الرموز والأسماء المختصرة بدلا من الحقول الطويلة . كما يجب عدم تكرار إدخـــال البيانات نظرا لان ذلك يجعل كفاءة هذه العمليات منخفضة.

أما **سهولة الاستخدام** فهي أيضا من المعايير الهامـــة لتقيـــم واجـــهات الاستخدام . وتعتبر سهولة الفهم والاستخدام وسهولة التعلم من الخصـــائص الـــتي يركز عليها المستخدم ، ويتم تجسيدها عند تصميم واجهات الاستخدام من خـــلال معرفة مستوى المستخدمين (ثقافتهم الحاسوبية) والعمليات المطلوب تنفيذها والبيئة التنظيمية لمكان العمل وظروفه . وكل ذلك يتم بالتعاون مع المســـتخدم . كمـــا يمكـــن اللجوء إلى أسلوب النماذج التجريبية للوصول إلى التصميم الأمثل لواجهات الاستخدام والتي تتوفر فيها الخصائص المذكورة أعلاه .

2– أنواع واجهات الاستخدام :

يمكن تصميم واجهات الاستخدام وفق الأساليب التالية :

– واجهات اللغة الطبيعية : Natural Language Interface

يتم التعامل مع الحاسوب من خلال هذه الواجهات باستخدام تعابير اللغات العادية (الطبيعية). وتعتبر هذه الواجهات مناسبة جداً للمستخدم الذي ليس لديه خبرة في استخدام الحاسوب. ويتم التحاور بين المستخدم والحاسوب من خلال رسائل تظهر على الشاشة ويقوم المستخدم بالإجابة عليها .

– واجهات الأسئلة والأجوبة Question and Answers :

يجري التحاور مع الحاسوب وفقاً لهذه الواجهات من خلال إظهار الحاسوب للسؤال فيقوم المستخدم بالإجابة عليه عن طريق لوحة المفاتيح . ولقد تطورت هذه الواجهات في بيئة تشغيل النوافذ و اصبحت بشكل صناديق حوار Dialogue Boxes يتم فيها الإجابة على عدد من الأسئلة في وقت واحد .

- القوائم Menus

وفقاً لهذا الأسلوب يقوم الحاسوب بإظهار عدد من الخيارات في قائمة ، ثم يقوم المستخدم بانتقاء أحد هذه الخيارات. وهكذا ينتقل المستخدم من قائمة إلى أخرى حتى الوصول إلى الوظيفة المطلوبة. ولقد تطورت هذه القوائم أيضاً في بيئة النوافذ بشكل ملحوظ وظهرت قوائم الخيارات المنسدلة Pull-Down Menus والقوائم الفجائية Pop-Up-Menus وغيرها.

– نماذج الإدخال والإخراج Input /Output Forms :

يتم الحوار بين المستخدم والحاسوب من خلال النماذج التي يظهرها الحاسوب فيقوم المستخدم بإملائها ثم يضغط زر موافق ليقوم الحاسوب بمعالجتها

وإظهار النتائج المطلوبة بشكل نموذج أيضاً. وهكذا يتم الحوار من خلال إدخال وإخراج المعلومات باستخدام هذه النماذج.

- واجهات لغات الأوامر Command - Language Interface :

يتم الحوار في هذه الواجهات بقيام المستخدم بإدخال الأوامر كما هو الحال في نظام التشغيل DOS ، فيقوم الحاسوب بالاستجابة وإظهار نتائج العمليات المطلوبة .

- واجهات التعامل المباشر Direct Manipulation Interface :

تعتمد هذه الواجهات على الاستخدام المكثف للرسوم البيانية ، كما هو الحال في بيئة التشغيل Windows التي تتضمن العديد من الأيقونات وعناصر التحكم المختلفة كالأزرار و القوائم وغيرها . كما يتم التعامل مع الحاسوب باستخدام هذه الواجهات ليس فقط من خلال لوحات المفاتيح بل أيضا باستخدام الفأرة وأجهزة القراءة الضوئية والماسحات وأجهزة التعرف الصوتي وغيرها . وتلاقي هذه الواجهات انتشاراً واسعاً في الأنظمة الحديثة بفضل التقدم التقني الكبير والسريع في هذا المجال .

3- تصميم التقارير Reports Design :

التقارير عبارة عن مخرجات مطبوعة تتضمن معلومات تلبي إحتياجات محددة . ولكي تكون هذه التقارير مفيدة لمستخدميها ، يجب أن تصمم بطريقة تضمن توفير المعلومات بالحجم والشكل والتوقيت والمكان المناسبين .

1.3 أنواع التقارير :

تتضمن أنظمة المعلومات تصميم أنواع متعددة من التقارير أهمها :

التقارير الداخلية Internal Reports :

وهي التقارير التي يتم استخدامها داخل المنظمة ، ولذلك فإن الاعتبـــارات الرئيسية عند تصميم هذه التقارير تنحصر في التكلفة الرخيصة واحتوائـــــها علـــى المعلومات اللازمة للمستخدمين .

التقارير الخارجية External Reports :

وهي التقارير الموجهة للأفراد والجهات الخارجية كالزبائن أو الموردیـــن أو الجهات الحكومية المختلفة . ويجب أن يراعي المصمم أن تكون هذه التقـــارير ذات محتوى مفيد وجودة مظهرها الذي يعكس صورة المنظمة ، وتقيدهـــا بالتعليمـــات والقوانين الحكومية الناظمة لها .

التقارير التفصيلية Detailed Reports :

تتضمن هذه التقارير بيانات تفصيلية عن الأنشطة المختلفة للمنظمة ، مثـــل موازين المراجعة ، وكشوفات حركة المواد وغيرها . وغالبا يتم إعداد هذه التقـــارير بهدف الاستخدام داخل المنظمة لمراجعة وتدقيق العمليات المختلفة . ولذلك فانـــــه يجب عند تصميمها مراعاة نفس الاعتبارات المتعلقة بتصميم التقارير الداخلية وهـــي المحتوى والتكاليف .

تقارير الإستثناءات Exceptional Reports :

وهي تقارير من نوع خاص يقوم نظام المعلومات بإصدارها في حالات خاصة عندما يكون من الضروري التدخل في عمل النظام. ولا تتضمن هــــذه التقاريـــــر عـــادة سوى بيانات محددة تتعلق بالمشكلة التي يتضمنـــــها التقريـــر. ويفضــــل أن

يلجأ المصمم إلى استخدام هذا النوع من التقارير باعتبارها تعتبر طريقة أكثر كفاءة لإصدار مخرجات النظام .

التقارير التلخيصية Summary Reports :

تقوم هذه التقارير بتجميع البيانات التفصيلية وإعدادها بشكل تقارير تتضمن بيانات إجمالية ملخصة عن أنشطة المنظمة . تعتبر هذه التقارير مهمة جداً للإدارة حيث توفر لها نتائج تحليل وتلخيص العمليات المختلفة وتعرض اتجاهات سير هذه العمليات.

تقارير حسب الطلب On-Request Reports :

يتم إصدار هذه التقارير عندما تظهر الحاجة إليها ، فمثلاً قد يحتاج المدير إلى تقرير بأسماء أفضل عشرين زبوناً للشركة ليصار إلى دعوتهم لحفل ما أو إرسال هدايا إليهم . ويجب على المصمم أن يوفر إمكانية إصدار هذه التقارير في النظام الذي يجري تصميمه .

2.3 الاعتبارات الضرورية لتصميم التقارير :

تطبع التقارير عادة على الورق ، ومن المهم اختيار حجم ونوع الورق المناسب عند تصميمها . بالنسبة للحجم يتم تحديده في ضوء حجم البيانات التي يمكن أن يتضمنها التقرير . أما من حيث النوع فيمكن استخدام الورق الأبيض العادي المستمر أو المقطع . كما يمكن استخدام النماذج مسبقة الطبع Preprinted Forms كما هو الحال في فواتير الهاتف والكهرباء وغير ذلك من الخدمات العامة. أما من حيث جودة الورق ومواصفاته فيجب استخدام ورق ابيض ذو

مواصفات جيدة للتقارير الموجهة إلى خارج المنظمة ، بينما يمكن استخدام ورق عادي للتقارير الداخلية .

وعند تصميم التقارير يجب أن يهتم المصمم باعتبارين هامين هما **الوضوح وسهولة الفهم** ، حيث لا فائدة من تقارير يحتاج مستخدمها إلى إعمال الفكر والتخمين لاستخلاص مضمونها، **والتكلفة المعقولة.**

يستخدم المصمم عند تصميم التقارير أدوات التصميم التالية : مخطط توزيع التقرير Printing Spacing Chart والنماذج التجريبية . ويساعد مخطط التقرير في ترتيب المعلومات بالشكل الذي يراه المصمم مناسباً للغرض المنشود من التقرير . حيث يجب أن يبدأ التقرير بمنطقة الرأس التي يجب أن يظهر فيها اسم الشركة واسم الإدارة صاحبة النظام ، وكذلك العنوان التفصيلي للتقرير ثم تاريخ التقرير ، يلي منطقة الرأس ، البيانات العامة أو الإجمالية المتعلقة بموضوع التقرير ، ثم يأتي جسم التقرير الذي يتضمن البيانات التفصيلية، ويبدأ بترويسة التقرير إذا كانت هذه البيانات التفصيلية مبوبة بشكل جدول . وينتهي التقرير عادة بمنطقة التذييل حيث تظهر فيها مجاميع البيانات وأرقام الصفحات وغير ذلك من البيانات ذات العلاقة ، كما هو مبين في الشكل (1.11)

اسم الشركة		التاريخ :
اسم الإدارة	**عنوان التقرير**	الوقت :
	وتتضمن ترويسة الجدول البيانات الإجمالية	
	جسم التقرير	
	تذييل التقرير	

شكل (1.11) ترتيب أقسام التقرير

وبفضل البرمجيات الجاهزة للحواسب الشخصية وكذلك تزايد استخدام لغات الجيل الرابع وأدوات هندسة البرمجيات بمساعدة الحاسوب، فإن الاتجاه يتزايد نحو استخدامها لتصميم التقارير وطباعتها ثم عرضها على المستخدم لإبداء ملاحظاته بشأنها واقتراح التعديلات اللازمة فيها . ويسمى هذا الأسلوب بالتصميم من خلال النماذج التجزيبية . ويتميز بكونه أكثر إنتاجية وفعالية ويسمح بالمشاركة الفعالة للمستخدمين في عملية التصميم.

4- تصميم الوثائق ونماذج الإدخال Source Documents :

يقصد بهذه الوثائق النماذج المستخدمة في تجميع المعلومات ثم إدخالها إلى الحاسوب . ومن أمثلتها العديد من الاستمارات التي يتم كتابة المعلومات فيها أثناء تنفيذ العمليات المختلفة (مثلا نماذج التسجيل في الجامعة أو أوامر الشراء أو نماذج المخزون، أو طلبات الاشتراك في المكتبة أو الجمعية أو غيرها). وتعتبر هذه الوثائق مهمة جدا في نظم المعلومات لكونها تزود النظام بالبيانات اللازمة لإنتاج المعلومات المطلوبة . ولذلك يجب أن يهتم المصمم بهذه الوثائق ويصممها بطريقة تضمن **سهولة فهمها وكتابة المعلومات** المطلوبة فيها، **ودون حدوث أي أخطاء** . كما يجب أن يتم تصميم هذه الوثائق أو النماذج بطريقة تتضمن **سرعة إدخال البيانات** الموجودة فيها إلى الحاسوب .

وغالبا تكون الوثائق المصدرية هذه مسبقة الطبع لتسهيل كتابة البيانات المطلوبة في الحقول المخصصة لذلك. كما يمكن أن تتضمن هذه الوثائق التعليمات اللازمة لكيفية كتابة البيانات في الحقول المختلفة . أما أهم اعتبارات تصميم هذه الوثائق فهي :

- المحافظة على بساطة التصميم لضمان سهولة الفهم والاستخدام .
- تسلسل قراءة الحقول من اليمين إلى اليسار (أو العكس إذا كانت باللغة الإنجليزية) ، ومن الأعلى إلى الأسفل) .
- يجب مراعاة الأمور المتعلقة بالتكلفة عند تصميم هذه الوثائق .
- تجميع الحقول ذات العلاقة في مجموعات وترتيبها وفق التسلسل المنطقي لاستخدامها.

- ترتيب الحقول الرقمية والحقول الأبجدية كل على حدة قدر الإمكان .
- استخدام العلامات والتأشير ما أمكن ذلك لتسهيل إدخال البيانات، مثلا طريقة التأشير على مربعات التدقيق Check Boxes .
- تخصيص فراغات كافية لكتابة البيانات المطلوبة .
- عند استخدام عدة نسخ يجب إعطاءها أرقاما وكتابة الوجهة التي يجب أن تستقر فيها هذه النسخ .
- يجب أن يتم ترتيب حقول البيانات المطلوب إدخالها إلى الحاسوب بنفس ترتيب هذه الحقول على النموذج الذي يظهر على الشاشة.
- تصميم النماذج بشكل يضمن تعبئتها بدقة وبسهولة وسرعة .

5- تصميم نماذج الشاشات Screen Forms :

تستخدم نماذج الشاشات لإدخال البيانات وإخراجها من النظام . ففي عمليات الإدخال يتم كتابة البيانات في هذه النماذج في الحقول المخصصة لها ، باستخدام لوحات المفاتيح، والفأرة أحيانا، وللتنقل بين هذه الحقول ، ويتم الحصول على هذه البيانات من الوثائق المصدرية التي تعرفنا عليها في الفقرة السابقة .

ولذلك يجب تصميم هذه النماذج بشكل يطابق أو يشبه إلى حد كبير الوثائق الأصلية التي ستقرأ منها هذه البيانات عند إدخالها . أما في عمليات الإخراج فإن نماذج الشاشات تعتبر طريقة فعالة جدا لإخراج البيانات التي لا يحتاجها المستخدم مطبوعة على الورق ، أي أنها تعتبر وسيلة لاسترجاع المعلومات. ونظرا لاستخدام نماذج الشاشات لإدخال البيانات ولإخراجها من النظام ، فإن الكثير من الاعتبارات الخاصة بتصميم التقارير وتصميم الوثائق المصدرية ، تنطبق تماما عليها

أيضا . وكذلك فإن أدوات تصميم هذه الشاشات هي نفسها المستخدمة في تصميم المخرجات والوثائق المصدرية . فبواسطة نماذج تصميم الشاشات Screen Layout Forms يقوم المصمم بتحديد التوزيع المناسب لحقول البيانات والعناصر المختلفة على الشاشة. ولكن تطور وانتشار برمجيات الحاسوب الشخصي، كما ذكرنا سابقا ، وأدوات هندسة البرمجيات بمساعدة الحاسوب ، ولغات الجيل الرابع ، جعل هذه المهمة أكثر إنتاجية وسهولة ، حيث يمكن تصميم الأشكال المختلفة لهذه النماذج وإجراء التعديلات والتغييرات فيها بسرعة وسهولة ، وعرضها على المستخدم ليراها بشكل فعلي وليس على الورق ، ويقدم ملاحظاته حولها . ولذلك فإن الأدوات التي ذكرناها أعلاه تعتبر مثالية لتصميم نماذج الشاشات بأسلوب النماذج التجريبية Prototyping . وعند تصميم الشاشات يجب أن يحاول المصمم الاستفادة قدر الإمكان من إمكاناتها المختلفة كالألوان والإضاءة والخطوط والعناصر الأخرى التي تبنى منها هذه النماذج .

ويتم إغناء نماذج الشاشات بأشرطة القوائم المنسدلة وأزار الضغط ومربعات التدقيق وغيرها الكثير من العناصر الأخرى التي تساعد المستخدم في الوصول إلى الوظائف المختلفة للنظام والتنقل إلى مكوناته بسرعة وسهولة .

وتطلق تسمية الواجهة التفاعلية Interactive Interface على مجموعة الرسائل التي يتم تبادلها بين المستخدم والحاسوب. ومن الطبيعي أن هذه الرسائل تتم من خلال الشاشات وتسمى هذه العملية بالحوار Dialogue بين المستخدم والحاسوب والذي يمكن أن يكون:

- حوار يتعلق بإدخال ومعالجة عملية معينة حدثت فعلا في النظام Transaction Processing Dialogue .

- حوار يتعلق بتصميم تقرير أو شاشة أو وثيقة مصدرية .
- حوار يتعلق باتخاذ قرار حول عمل ما ، مثلا البحث عن حل لمشكلة معينة .
- حوار يتعلق بالاتصال والتنسيق مع الجهات الأخرى داخل أو خارج المنظمة .

في جميع أنواع الحوارات المذكورة أعلاه تستخدم القوائم Menus كأسلوب بسيط وسهل للتعامل بين المستخدم والحاسوب ، حيث يعرض الحاسوب عددا من الخيارات ، ويقوم المستخدم باختيار المطلوب ، وبعد سلسة من هذه الاختيارات يتم الوصول إلى العمل المطلوب.

أما الأسلوب الآخر لتصميم الحوارات فيتم من خلال مربعات الحوار المشار إليها أعلاه حيث يظهر الحاسوب مربعا محددا على الشاشة، ويطلب من المستخدم إدخال عدد من الخيارات مرة واحدة، وكذلك تحديد البيانات اللازمة لتنفيذها . ولقد ساعد ذلك في جعل الحوارات أسهل وأسرع وأكثر كفاءة. فبدلا من التنقل بين سلسة من القوائم يمكن الوصول في خطوة واحدة إلى العمل المطلوب.

لقد وفرت بيئات التشغيل الحديثة أدوات وإمكانات جديدة جعلت التفاعل بين المستخدم والحاسوب في غاية السهولة والكفاءة كالقوائم المنبثقة والقوائم المنسدلة والنوافذ المتعددة والأيقونات وغيرها من عناصر الواجهات البيانية، وكذلك استخدام الوسائط المتعددة (الصوت والصور والفيديو والكلمات المنطوقة وغيرها الكثير)، مما يفتح الباب واسعا أمام المصمم لاختيار ما يرغبه من هذه العناصر التصميمية، ويجعل عملية تصميم واجهات الاستخدام تتطلب مهارة وخبرة كبيرتين.

وأخيرا فان توفر إمكانات المساعدة الحساسة للسياق Context-Sensitive Help تعتبر من العناصر الهامة لتصميم واجهات حوار تفاعلية ناجحة.

أسئلة الفصل:

1- عرف واجهات الاستخدام واشرح الخصائص التي يجب توفرها فيها؟

2- اشرح الأنواع المختلفة لواجهات الاستخدام؟

3-. اشرح أهمية التقارير وأنواعها؟

4- اشرح الاعتبارات الضرورية التي يجب مراعاتها عند تصميم التقارير؟

5- ما هي أهمية وثائق ونماذج الإدخال وما هي الاعتبارات اللازمة لتصميمها؟

6- اشرح الاستخدامات المختلفة لنماذج الشاشات؟

7- اشرح الواجهة التفاعلية وأنواع الحوارات وأساليب تنفيذها؟

الفصل الثاني عشر

تصميم قواعد البيانات

1- أهداف ومراحل تصميم البيانات :

يهدف تصميم قاعدة بيانات إلى تحديد المواصفات التفصيلية لقاعدة البيانات اللازمة للنظام . ففي مرحلة التحليل يتم إعداد نموذج بيانات النظام الحالي باستخدام مخطط الكينونة – العلاقة، أو غير ذلك من المخططات التي تمت دراستها في الفصل السابع. ثم في مرحلة التصميم العام يتم تدقيق ومراجعة هذا المخطط في ضوء النموذج المنطقي للنظام الجديد ، الذي يتم التوصل إليه في تلك المرحلة . ويعتبر مخطط الكينونة – العلاقة المعدل جزءاً من هذا النموذج المنطقي للنظام الجديد . وفي مرحلة التصميم التفصيلي يتم استناداً إلى هذا المخطط تحديد المتطلبات المتعلقة بالبيانات اللازمة لعمل النظام .

عند تصميم قاعدة بيانات النظام يجب مراعاة المبادئ التالية :

- **التحكم بتكرار البيانات** ليكون في اقل حدود ممكنة ، والحالة المثالية هي تخزين عنصر البيانات في مكان واحد فقط .
- **كفاءة الأداء في قاعدة البيانات** ، وهذا يعني توفير البنية والأساليب التي تضمن سرعة الوصول إلى البيانات الموجودة في القاعدة واسترجاعها

- **توفير المرونة** اللازمة لتعديل قاعدة البيانات . وهذا يتضمن أيضاً إجراء التعديلات اللازمة في بنية قاعدة البيانات وليس فقط في محتوياتها .
- **الحماية والسلامة** : وهذا يعني تصميم الإجراءات اللازمة لحماية قاعدة البيانات من جميع الأخطار المحتملة، ووضع الخطط اللازمة لذلك .

وبشكل عام فإن تصميم قاعدة بيانات النظام يتم في مرحلتين رئيسيتين هما:

أ) **التحليل العلاقاتي Relational Analysis للبيانات** ، بهدف الإقلال من التكرارات، وتحسين بنية نموذج بيانات النظام . وتسمى هذه المرحلة إعداد النموذج العلاقاتي للنظام Relational Modeling .

ب) **تصميم قاعدة البيانات اللازمة للنظام** : وذلك استناداً إلى النموذج العلاقاتي الذي تم التوصل إليه في المرحلة السابقة. وفي نهاية هذه المرحلة يتم التوصل إلى التصميم المادي لقاعدة البيانات Physical Database Design ، الذي يستخدم في مرحلة التنفيذ لإنشاء قاعدة البيانات .

وسندرس فيما يلي أهم الأنشطة التي يجب القيام بها خلال المرحلتين المذكورتين أعلاه .

2- التحليل العلاقاتي : Relational Analysis

يهدف هذا التحليل إلى تحسين شكل وبنية مخطط الكينونة – العلاقة، أي نموذج بيانات النظام ، والإقلال ما أمكن من التكرارات فيه . ولقد أطلق على هذا

التحليل صفة " العلاقاتي " لكونه يستند إلى مفهوم العلاقة Relation التي يقصد بها جدول أو قائمة من القيم المختلفة .

أما أهم الخطوات التي تتضمنها مرحلة التحليل العلاقاتي فهي :

1) تحويل مخطط الكينونة - العلاقة في النموذج المنطقي الجديد إلى مجموعة العلاقات Set of relations المكونة له .

2) تنميط البيانات Data Normalization : أي تحويل العلاقات التي تم التوصل إليها أعلاه إلى علاقات طبيعية Normal Relations.

وسنقدم فيما يلي شرحاً مفصلاً لهاتين الخطوتين:

الطلاب

رقم الطالب STUDENT-ID	اسم الطالب STUDENT-NAME	الهاتف TELEPHONE
A 120	محمد محمود على	5155411
A 125	ابراهيم الأفقى	5223470
E 140	سعيدة الواسطى	4617733
N020	أحمد عليان	5355212
N051	سميرة ابراهيم	4662280

يدرسون

رقم الطالب STUDENT-ID	رقم المقرر COURSE-ID	الفصل الدراسى SEMESTER
A 120	1131	1/1998
A 125	6400	2/1998
E 140	1120	1/1998
N020	1245	1/1998
N051	1131	2/1998

المقررات

رقم المقرر COURSE-ID	اسم المقرر COURSE-NAME	الساعات المعتمدة CREDET-HOURS
1120	البرمجة بلغة C	3
1131	أنظمة المعلومات الحاسوبية	3
1245	مبادئ علم الحاسوب	2
6499	ثقافة عربية وإسلامية	2

الطلاب
رقم الطالب
اسم الطالب
الهاتف
يدرسون
رقم الطالب
رقم المقرر
الفصل الدراسي
المقررات
رقم المقرر
إسم المقرر
الساعات المعتمدة

شكل (1.12) العلاقات ومخطط الكينونة–العلاقة

1.2 تحويل مخطط الكينونة – العلاقة إلى نموذج علاقاتي:

لقد ذكرنا أعلاه أن العلاقة Relation هي عبارة عن جدول أو قائمة تمثل أسطره (صفوفه) السجلات المتعلقة بالكينونات المختلفة التي يتم وصفها في الجدول ، أما أعمدته فهي عبارة عن **الخصائص أو الصفات** Attributes المستخدمة لوصف هذه الكينونة . ويتم الإشارة إلى العلاقات من خلال إعطاءها تسميات فريدة تعبر عن محتوياتها .

فمثلاً الجداول المبينة في الشكل (1.12) هي عبارة عن علاقات تمثل صفوفها (أسطر الجدول) **سجلات أو حدوثات Tuples** كما تمثل أعمدتها **خصائص أو صفات العلاقة Attributes** . في الجداول المبينة في الشكل (1.12) أعلاه نجد مثلاً أن جدول الطلاب يتضمن ثلاث أعمدة هي رقم الطالب واسم الطالب والهاتف ، تسمى هذه الأعمدة بخصائص أو صفات العلاقة . ويمكن التعبير عن هذا الجدول بشكل رمز علاقة تكتب على النحو التالي :

STUDENTS (<u>STUDENTS-ID</u>, STUDENT-NAME, TELEPHON)

وهكذا نجد أنه يتم تمثيل العلاقة بكتابة اسمها يليه فوراً قوسان يسرد ضمنهما أسماء خصائص Attributes هذه العلاقة . ونلاحظ وجود خط تحت بعض هذه الخصائص التي تمثل مفتاح العلاقة Relation Key . وذلك بنفس الطريقة المستخدمة للإشارة إلى معرف المجموعة في مخططات الكينونة –العلاقة. وبذلك يمكن تمثيل الجداول الثلاث الموجودة في الشكل (1.12) أعلاه بشكل رموز علاقات على النحو التالي :

STUDENTS (<u>STUDENT-ID</u>, STUDENT-NAME, TELEPHON)
STUDY (<u>STUDENT-ID, COURSE-ID</u>, SEMESTER)

COURSES (<u>COURSE-ID</u>, COURSE-NAME, CREDIT-HOURS)

ويجب أن يكون للعلاقة **اسم فريد** ضمن النظام ، كما يجب أن تكون للخصائص أيضاً تسميات فريدة أيضاً . وكل سطر في العلاقة يمثل أو يصف **كينونة واحدة ضمن العلاقة** (طالب واحد أو مقرر واحد) ، أما ترتيب الأسطر والأعمدة ضمن العلاقة فليس ذي قيمة. ولكن لا يمكن أن تتضمن العلاقة سطران متماثلان تماماً ، ولذلك فإن لكل علاقة مفتاحاً يمكن أن يكون خاصية واحدة أو عدة خصائص تضمن أن تكون قيم البيانات الموجودة في السجل الواحد فريدة غير متكررة . ولكي تكون العلاقة طبيعية يجب أن تكون بسيطة، أي تحتوي قيماً بسيطة فقط. تسمى العلاقات التي تحتوي قيما مركبة Group Values علاقات غير طبيعية و يجب تحويلها إلى علاقات طبيعية بتجزئه أو تقسيم القيم المركبة الموجودة فيها إلى قيم بسيطة .

إن تحويل مخطط الكينونة – العلاقة إلى مجموعة العلاقات Set of relations يعتبر خطوة أولى نحو الانتقال من هذه المخططات البيانية نحو تحديد مجموعة الملفات اللازمة لتخزين بيانات النظام .فكل جدول أو علاقة سيتم تجسيدها مادياً بشكل ملف بيانات ، سجلاته عبارة عن اسطر الجدول (العلاقة) وحقوله هي الخصائص التي تتضمنها العلاقة (أعمدة الجدول). والغاية أو الهدف الرئيسي من هذا التحويل هو القيام بتحليل هذه العلاقات وتحويلها إلى **علاقات طبيعية** Normal Relations لا تحتوي إلا الحد الأدنى من التكرارات . فنموذج البيانات الجيد أو الأمثل هو ذلك النموذج الذي تكون جميع العلاقات المكونة له طبيعية . وتسمى هذه العملية بالتطبيع أو التسوية أو التنميط Normalization التي يتم من خلالها تحويل جميع علاقات النموذج إلى علاقات طبيعية .

2.2 تطبيع البيانات Normalization :

لقد ذكرنا أعلاه أنه لكي تكون العلاقات طبيعية يجب أن تحتوي قيما بسيطة غير مركبة . فإذا احتوت العلاقة (أو الجدول) على قيم مركبة يكون شكلها غير طبيعيا Non-Normal Form Relation . ولذلك يجب تحويلها لتصبح طبيعية ، وتسمى عملية التحويل هذه **تطبيع أو تسوية** Normalization ، ويمكن من خلالها تحسين هيكل بيانات النظام ليكون أكثر كفاءة. وتتم عملية التطبيع هذه بشكل تدريجي وفق عدد من المستويات هي:

الشكل الطبيعي الأول للعلاقة : First Normal Form Relation

تكون العلاقة في الشكل الطبيعي الأول إذا لم يكن فيها حقول بيانات متعددة القيم Multivalued . أما إذا احتوت العلاقة حقول ذات قيم متعددة فإنه يجب تقسيمها إلى علاقتين ، على النحو المبين في المثال التالي :

لنفرض أنه لدينا أمر الشراء الذي يحوي البيانات التالية :

– رقم أمر الشراء	PO-NO
– تاريخ أمر الشراء	PO-DATE
– رمز المورد .	VENDOR-NO
– اسم المورد	VENDOR-NAME
– عنوان المورد	VENDOR-ADDRESS
– {رمز المادة	{ ITEM-CODE
– { اسم المادة	{ ITEM-DESCRIPTION
– { الكمية	{ ITEM-QTY

- { السعر   { ITEM-PRICE
- { القيمة للفقرة   { ITEM-VALUE
- { القيمة الإجمالية   { PO-AMOUNT

إن الفقرات الموضوعة بين قوسين يمكن تكرارها في أمر الشراء عدة مرات (يمكن أن يكون لها قيم متعددة في أمر الشراء) لأن أمر الشراء الواحد يحتوي عدة أسطر لسرد الفقرات المطلوب شراءها .

يمكن كتابة العلاقة التي تمثل أمر الشراء على النحو التالي :

PURCHES-ORDERS (PO-NO, PO-DATE, VENDOR-NO, VENDOR-NAME, VENDOR-ADDRESS, { ITEM-CODE, ITEM-DESCRIPTION, ITME-QTY,ITEM-PRICE, ITEM-VALUE}, PO-AMOUNT)

تسمى العلاقة أعلاه غير طبيعية non-normal relation لكونها تتضمن فقرات بيانات متعددة القيمة Multivalued، هي عناصر أسطر المواد التي يمكن أن يتضمنها أمر الشراء . ولكي يتم تحويل هذه العلاقة أعلاه إلى الشكل الطبيعي الأول يجب فصل الحقول (الخصائص متعددة القيمة ووضعها في علاقة مستقلة يتكون مفتاحها من مفتاح العلاقة الأصلية، أي PO-NO والمفتاح الخاص بتلك البيانات المتكررة أي ITEM-CODE.

وهكذا يصبح لدينا علاقتين في الشكل الطبيعي الأول ، أي لا تحتويان بيانات متكررة أو متعددة القيمة وهما :

- PURCHASE-ORDERS(PO-NO, PO-DATE, VENDOR-NO, VENDOR-NAME, VENDOR-ADDRESS, PO-AMOUNT)
- PURCHASED-ITEMS (PO-NO, ITEM-CODE, ITEM-DESCRIPTION, ITEM-QTY, ITEM-PRICE, ITEM-VALUE)

الشكل الطبيعي الثاني للعلاقة Second Normal Form Relations

لكي تكون العلاقات في الشكل الطبيعي الثاني ، يجب أن ترتبط جميع خصائص (صفات أو حقول) العلاقة بكامل المفتاح المركب، أي بكل عناصره وليس بجزء منها فقط . فمثلاً يمكن أن نلاحظ في العلاقة التي تمثل أسطر أمر المشتريات PURCHASED-ITEMS أن اسم المادة ITEM-DESCRIPTION لا يتعلق إلا بجزء من المفتاح المركب لهذه العلاقة وهو رمز المادة ITEM-CODE. بينما نلاحظ أن الكمية المطلوبة ITEM-QTY والسعر ITEM-PRICE يرتبطان بكلا الجزئين المكونين للمفتاح المركب لكونهما يعبران عن الكمية المطلوبة والسعر لهذه المادة في أمر الشراء هذا . ونظراً لوجود مثل هذه الارتباطات الجزئية فإن العلاقة الثانية المبينة أعلاه ليست في الشكل الطبيعي الثاني. ولكي تصبح كذلك يجب أيضاً فصل تلك العناصر التي ترتبط بجزء من المفتاح المركب ووضعها في علاقة مستقلة يكون مفتاحها هو نفس الجزء الذي ترتبط به من المفتاح المركب . وبذلك يمكن كتابة العلاقة PUCHASED-ITEMS بالشكل الطبيعي الثاني على النحو التالي :

- PUCHASED-ITEMS (PO-NO, ITEM-CODE, ITEM-QTY, ITEM-PRICE, ITEM-VALUE)
- ITEMS (ITEM-CODE, ITEM-DESCRIPTION)

كما أنه لدى تفحص العلاقة المسماة PUCHASE-ORDERS نجد أن الحقول أو الخصائص المتعلقة بالمورد وهي VENDOR-ADDRESS,VENDOR-NAME ليس لها علاقة مباشرة برقم أمر الشراء PO-NO ولذلك يمكن وضعها في علاقة منفصلة . أي يمكن تقسيم علاقة أوامر الشراء أيضاً إلى علاقتين من الشكل الطبيعي الثاني .

وهكذا يصبح لدينا أربعة علاقات في الشكل الطبيعي الثاني وهي :

- VENDORS (<u>VENDOR-NO</u>, VENDOR-NAME, VENDOR-ADRESS)
- PURCHASE-ORDERS(<u>PO-NO</u>,PO-DATE,VENDOR-NO,PO-AMOUNT)
- PURCHASED-ITEMS(<u>PO-NO,ITEM-CODE</u>, ITEM-QTY,ITEM-PRICE, ITEM-VALUE)
- ITEMS (<u>ITEM-CODE</u>, ITEM-DESCRIPTION)

د

الشكل الطبيعي الثالث للعلاقة : 3NF Relations

في الشكل الطبيعي الثالث للعلاقة يجب أن لا تكون الحقول غير المفتاحية مرتبطة ببعضها البعض ، أي لا يمكن اشتقاق قيمة العناصر أو الحقول غير المفتاحية non-key-attributes من بعضها البعض . فمثلا إذا تأملنا العلاقة الثالثة أعلاه والمسماة PURCHASED-ITEMS نجد أن الخاصية أو العنصر المسمى " قيمة الفقرة " ITEM-VALUE يمكن احتسابها من الفقرات الأخرى في هذه العلاقة وذلك بضرب الكمية ITEM-QTY في السعر ITEM-PRICE. ولذلك يمكن حذف هذه الفقرة من هذه العلاقة. وبذلك تكون العلاقات التالية جميعا وفق الشكل الطبيعي الثالث وهي :

- VENDORS (<u>VENDOR-NO</u>, VENDOR-NAME, VENDOR-ADDRESS)
- PURCHASED-ORDERS (PO-NO, PO-DATE, VENDOR-NO, PO-AMOUNT)
- PURCHASED-ITEMS (PO-NO,ITEM-CODE, ITEM-QTY,ITEM-PRICE)
- IEMS (ITEM-CODE,ITEM-DESCRIPTION)

3. تصميم قاعدة بيانات النظام Database design

في هذه المرحلة يتم إعداد التصميم المنطقي ثم التصميم المادي لقاعدة البيانات اللازمة لنظام المعلومات الذي يجري تصميمه . تبدأ هذه المرحلة بعد التوصل إلى البنية المثلى لبيانات النظام ، الناتجة عن عملية التطبيع Normalization التي درسناها في الفقرة السابقة .

وتتضمن عملية تصميم قاعدة بيانات النظام الخطوات الرئيسية الثلاث التالية:

1) **تحويل مخطط الكينونة – العلاقة** ، الذي تم تحسينه في المرحلة السابقة، إلى بنية سجلات منطقية Logical Record Structure(LRS).

2) **التصميم المنطقي لقاعدة البيانات** : تحويل البنية المنطقية للسجلات ، التي تم التوصل إليها في الخطوة السابقة إلى البنية المنطقية لنظام إدارة قواعد البينات Logical DBMS Structure، والتي يمكن أن تكون هرمية Hierarchical أو شبكية Network أو علاقاتية Relational.

3) **التصميم المادي لقاعدة البيانات** : تحويل البنية المنطقية لنظام إدارة قاعدة البيانات إلى بنية مادية أي تحديد قاعدة بيانات النظام .

علماً أن الخطوة الأولى تتم قبل إختيار نظام إدارة قاعدة البيانات ولذلك فإنها توصف بكونها DBMS independent، أما الخطوتان الثانية والثالثة فهما تتمان استنادا إلى نظام إدارة قواعد بيانات محدد ، ولذلك توصف هذه العمليات بكونها DBMS dependent. وسنشرح فيما يلي الخطوات الثلاث أعلاه اللازمة لتصميم قاعدة البيانات:

1.3 إنشاء البنية المنطقية للسجلات:

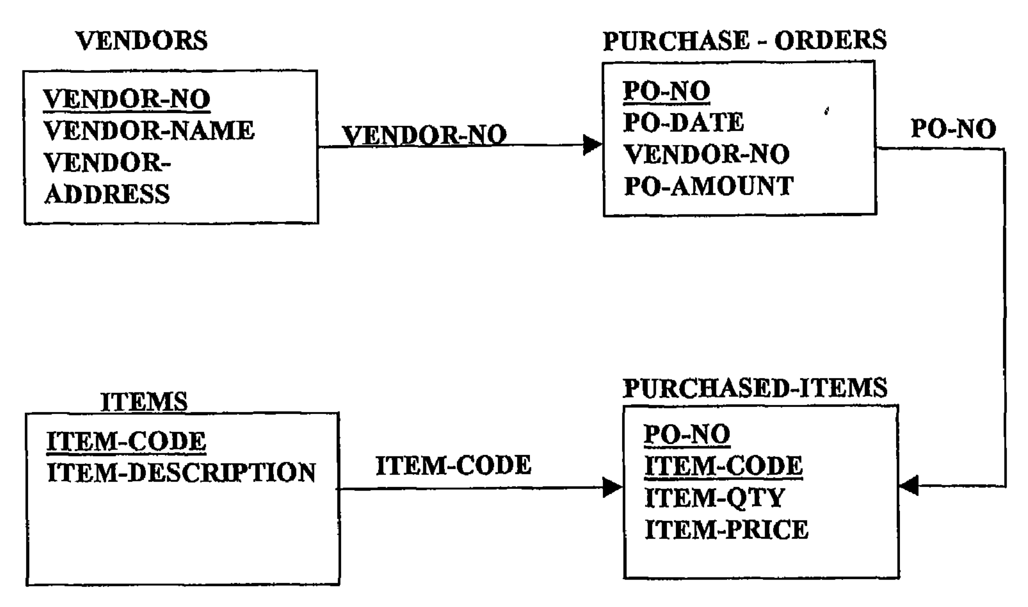
تتضمن هذه الخطوة تحويل نموذج البيانات أي مخطط الكينونة – العلاقة أو النموذج العلاقاتي لبيانات النظام ، الذي تم التوصل إليه في المرحلة السابقة (مرحلة التحليل العلاقاتي) إلى ما يسمى بالبنية المنطقية للسجلات (LRS) التي تتضمن تحديد:

- أنواع السجلات Records types اللازمة لتخزين بيانات النظام .
- متطلبات الوصول إلى هذه السجلات Accessing Requirements.
- بيانات كمية مثل حجم الحقول المختلفة وعدد السجلات في الملف ومعدل تكرار أو تواتر عمليات الوصول إلى هذه السجلات Access frequency وغيرها.

تعتبر البنية المنطقية للسجلات بما تتضمنه من معلومات على النحو المبين أعلاه ، توصيفا لقاعدة البيانات اللازمة للنظام Database Specification. وفي ضوء هذا التوصيف يتم إختيار نظام إدارة قواعد البيانات DBMS الذي سيستخدم لبناء وإدارة قاعدة بيانات النظام .

تتألف البنية المنطقية للسجلات (LRS) عادة من عدة أنواع مختلفة من السجلات التي يتم تمثيلها بشكل مستطيلات يكتب فوقها تسمياتها ، ويكتب في داخلها أسماء فقرات البيانات التي يتكون منها كل نوع ، كما هو مبين في الشكل (2.12). وهذا يعني أن كل نوع من السجلات يتألف من مجموعة من الحقول أو عناصر البيانات، ويتم تمثيل العلاقات بين الأنواع المختلفة للسجلات بشكل أسهم تسمى صلات أو روابط Links ، يكتب فوقها أسماء الحقول المشتركة بين الأنواع المختلفة للسجلات كما هو واضح من الشكل (2.12) المشار إليه أعلاه .

تسمى صلات أو روابط Links ، يكتب فوقها أسماء الحقول المشتركة بين الأنـــواع المختلفة للسجلات كما هو واضح من الشكل (2.12) المشار إليه أعلاه .



شكل (2.12) مثال للبنية المنطقية للسجلات

ونلاحظ في البنية المنطقية للسجلات أن اتجاه الأسهم يكون من الســــجل المالك Owner إلى السجل التابع . فمثلا ينطلق السهم الذي يمثل رقم المورد من نوع السجل VENDORS وهو المالك إلى نوع الســـجل PURCHASE-ORDERS. وكذلك الحال بالنسبة للسهم الذي يمثل الحقل المشترك "رقم أمر الشـــراء " فـــهو ينطلق من السجل المالك "أوامر الشراء" إلى السجلات التابعة " المـــواد المشـــتراة" PURCHASED-ITEMS.

أما السهم أو العلاقة ITEM-CODE فهي تنطلق مـــن الســـجل المـــالك ITEMS إلى السجلات التابعة والسماة "المواد المشتراة". وأخيرا تجدر الإشارة إلى أنه

يمكن إنشاء البنية المنطقية للسجلات انطلاقا من النموذج العلاقاتي Relational model لبيانات النظام ، أو من مخطط الكينونة – العلاقة E-RD . وهذا ما سنقوم بشرحه فيما يلي :

1.1.3 إنشاء البنية المنطقية للسجلات انطلاقا من النموذج العلاقاتي للبيانات :

وفقا لهذه الطريقة يتم تحويل النموذج العلاقاتي لبيانات النظام إلى البنية المنطقية للسجلات ، وذلك من خلال الخطوتين التاليتين:

1) تحديد أنواع السجلات :

وهذا يعني تحويل كل علاقة Relation إلى نوع سجل Record type ورسمه بشكل مستطيل وكتابة اسم العلاقة فوقه ، ثم كتابة أسماء الخصائص (الحقول) داخل المستطيل الذي يمثل نوع السجل كما هو مبين في الشكل (2.12). وهكذا يتم تحويل كل علاقة إلى نوع سجل ، وتصبح خصائص العلاقة عبارة عن الحقول Data Fields التي يتضمنها ذلك النوع من السجلات .

2) تحديد الروابط (الصلات) اللازمة لتمثيل العلاقات بين الأنواع المختلفة للسجلات :

يتم تحديد العلاقات بين الأنواع المختلفة للسجلات من خلال استخدام مفهوم المفاتيح الأجنبية Foreign Keys، ويقصد بالمفتاح الأجنبي خاصية أو مجموعة من الخصائص (الحقول) موجودة في علاقة ما (ولكنها ليست مفتاح هذه العلاقة) والمستخدمة في علاقة أخرى كمفتاح لتلك العلاقة . فمثلا نجد أن رمز المورد في الشكل (2.12) هو أحد خصائص العلاقة " أوامر الشراء " وهو عنصر غير مفتاحي فيها ، ولكنه يستخدم في علاقة أخرى وهي " الموردون " كمفتاح لتلك

العلاقة . ولذلك يقال عن الحقل VENDOR-NO الموجود في علاقة أوامر الشــــراء بأنه مفتاح أجنبي. وهذا المفتاح الأجنبي هو الذي يتم استخدامه للربط بين نوعـــي السجلات " الموردون " و " أوامر الشراء " وهكذا .

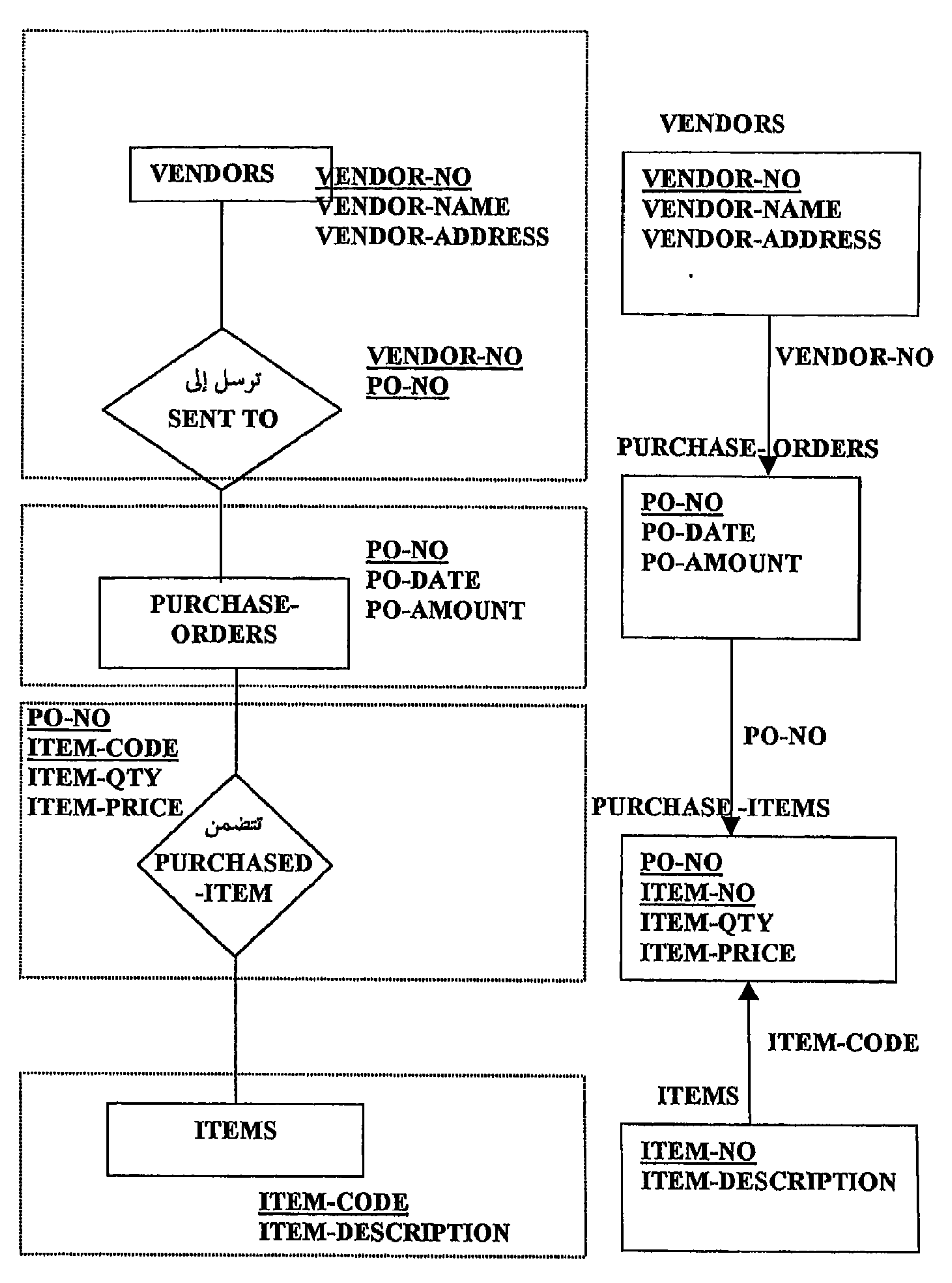
ومن الواضح أن اتجاه العلاقة يكون من نوع السجل الذي يحوي المفتـــاح الرئيسي (المالك) إلى نوع السجل الذي يحوي المفتاح الأجنبي .

2.1.3 إنشاء البنية المنطقية للسجلات انطلاقاً من مخطط الكينونة – العلاقة:

وفقاً لهذه الطريقة يتم تحويل مخطط الكينونة – العلاقـــة E-RD إلى بنيـــة منطقية للسجلات اللازمة للنظام وفق الخطوتين التاليتين :

1) **تحديد أنواع السجلات المنطقية** اللازمة للنظام وذلك بتحويـــل كـــل مجموعة كينونة أو مجموعة علاقة إلى نوع سجل منطقي Logical Record Type . ويمكن ضم أو دمج أنواع السجلات إذا كانت لها نفس المفـــاتيح (الصفات المفتاحية) كما هو مبين في الشكل (3.12) .كما يتم تحويـــل مجموعة الكينونات التابعة إلى سجلات منطقية .

2) **تحديد الروابط أو الصلات Links** : ينطلق اتجاه الأسهم التي تمثلها هذه الروابط من السجلات المنطقية التي تمثل مجموعة الكينونات (أوامر الشراء) نحو السجلات المنطقية التي تمثل مجموعة العلاقات (المـــواد المشـــتراه) . كمـــا هو مبين في الشكل (3.12).



شكل (3.12) تحويل مخطط الكينونة العلاقة E-RD الهرمية بنية سجلات منطقية (LRS)

وكذلك يتم ربط السجلات المنطقية التي تمثل مجموعة الكينونات التابعة مع السجلات التي تمثل مجموعة الكينونات ألام ، وينطلق اتجاه السهم الذي يمثل هـذه الصلة من السجل المنطقي لمجموعة الكينونة ألام نحو السجل المنطقي الـذي يمثـل مجموعة الكينونات التابعة .

وبعد الانتهاء من إعداد المخطط الذي يمثل البنية المنطقية للسـجلات يتـم تحديد مواصفات قاعدة البيانات Database Specification والتي تتضمن بالإضافـة إلى البنية المنطقية للسجلات ، المعلومات التالية:

أ) **متطلبات الوصول إلى البيانات** : ويتم من خلالها تحديد طرق الوصـول Accessing Paths اللازمة لاسترجاع البيانات من القاعدة ، وهذا يتضمن تحديد :

- الطلبات الممكنة لاسترجاع البيانـات Requests وأنـواع السـجلات المنطقية التي يجب الوصول إليها لتلبية كل طلب .
- التسلسل الذي سيتم وفقه الوصول إلى الأنواع المختلفة من السـجلات المنطقية ، وهذا يسمى تسلسل الوصول الى السجلات المنطقية .
- المفاتيح المستخدمة للوصول إلى السجلات.
- فقرات البيانات (الحقول) التي سيتم استرجاعها من كل نـوع مـن السجلات.
- عدد السجلات التي سيتم الوصول إليها.

ب) **بيانات كمية** تساعد في إعطاء فكرة **عن حجم البيانـات وحجـم عمليات** الوصول إلى قاعدة البيانات ، وذلك مثل حجم الحقـول في السـجلات

المختلفة، وعدد السجلات الممكنة ، وتكرار أو دورية الوصـــول إلى الســـجلات وهكذا.

2.3 التصميم المنطقي لقاعدة البيانات :

بعد الانتهاء من إعداد مواصفات قاعدة البيانات على النحو المشار إليـــه في الخطوة السابقة ، يتم في هذه الخطوة تحديد التصميم المنطقـــي المناســـب لقـــاعدة البيانات وفق المواصفات المحددة . وعندما يكون النظام بســـيطاً وصغـــيراً يمكـــن استخدام طريقة الملفات للإحتفاظ ببيانات النظام ، أما عندما يكون النظام كبـــيراً ومعقداً ، أو تكون متطلبات الوصول إلى البيانات عالية المســـتوى فانـــه يفضـــل استخدام أسلوب قواعد البيانات. ويتعاظم التوجه حالياً نحـــو اســـتخدام قواعـــد البيانات للاحتفاظ ببيانات النظام باعتبارها أسلوبا اكثر تطوراً يوفر العديد من المزايا أهمها تقليل التكرار إلى أدنى حد ممكن ، وتوفير استقلالية البيانات عـــن الـــبرامج، بالإضافة إلى إمكانات الحماية والرقابة وتدقيق الأخطاء وغيرها .وبالرغم مـــن أن أسلوب الملفات يعتبر الآن تقليدياً إلا أننا سنقدم فيما يلي شرحاً لكلا الأســـلوبين : الملفات وقواعد البيانات .

1.2.3 تحويل البنية المنطقية للسجلات إلى مجموعة الملفـــات اللازمة للنظام :

أ) أنواع الملفات :

يمكن أن يتضمن نظام المعلومات الذي يجري تطويره أنواعاً مختلفـــة مـــن الملفات أهمها :

الملفات الرئيسية Master files :

وتحتوي عادة البيانات الأساسية المتعلقة بالكينونات الرئيسية للنظام مثل ملفات الزبائن والموردين والطلاب والموظفين والآلات والمنتجات وغيرها .

ملفات العمليات Transaction files :

ويتم في هذه الملفات تسجيل بيانات الأحداث أو الوقوعات أو الحركات التي يمكن أن تتم بشكل متكرر مثل عمليات المبيعات أو حركة المنتجات أو حركة مواد المخزون أو أوامر الشراء أو غيرها من عمليات النظام . وتستخدم هذه الملفات لتحديث بعض حقول الملفات الرئيسية ولإنتاج تقارير تلخيصية عن العمليات المختلفة في المنظمات .

الملفات المؤقتة Temporary files :

تستخدم هذه الملفات للاحتفاظ بالبيانات الناتجة عن عمليات المعالجة لفترة قصيرة ، ريثما يتم تحويلها إلى المخرجات المطلوبة. ومن الأمثلة على ذلك ملفات البحث أو الفرز التي يتم فيها تخزين نتائج عمليات البحث أو الفرز بشكل مؤقت أثناء إجراء هذه العمليات . ثم بعد الانتهاء من البحث أو الفرز يتم تحويل البيانات الموجودة في هذه الملفات المؤقتة إلى ملفات التقارير أو المخرجات ، ثم يتم بعد ذلك، حذف هذه الملفات المؤقتة.

الملفات المعلقة Suspense file :

وتستخدم لتسجيل الأخطاء التي يتم اكتشافها من قبل برامج التدقيق المختلفة . وهي أيضا تشبه الملفات المؤقتة حيث يتم حذفها بعد دراستها وتصحيح الأخطاء التي تم اكتشافها .

الملفات الجدولية Table files :

تستخدم هذه الملفات لتخزين البيانات الثابتة التي يتم الرجوع إليها بشكل متكرر من قبل البرامج التطبيقية المختلفة . ومن أمثلة هذه الملفات : ملفات الرسائل - ملفات الرموز المختلفة كالقرى والمدن والدول والمنتجات وغيرها .

الملفات الاحتياطية Backup files :

وهي عبارة عن نسخ من جميع ملفات النظام (وبشكل خاص ملفات البيانات)، يتم إنشاءها بصورة دورية ، والاحتفاظ بها كإجراء احتياطي لتستخدم عند وقوع الأعطال التي يمكن أن تؤدي إلى فقدان الملفات الأصلية .

ب) خطوات تصميم الملفات :

عند استخدام أسلوب الملفات للاحتفاظ ببيانات النظام، يمكن اتباع الخطوات التالية:

1- تحديد وتعريف جميع الملفات اللازمة للنظام وإعطاءها تسميات فريدة ذات دلالة، وذلك بتحويل كل نوع سجلات إلى ملف : وهكذا ففي المثال المبين في الشكل (3.12) يكون مطلوباً إنشاء أربعة ملفات هي :

- ملف الموردون VENDORS FILE.
- ملف أو الشراء PURCHASE-ORDERS FILE .
- ملف المواد المشتراه (تفاصيل أو الشراء) PURCHASED ITEMS FILE .
- ملف المواد ITEMS FILE.

2- تحديد نوع كل ملف: رئيسي أو عمليات أو غير ذلك.

3- وضع قائمة بجميع عناصر البيانات التي سـيتم تخزينـها في سـجلات الملفات . وتحديد حجم كل عنصر ونوع بياناته .ويتم ذلك بـالرجوع إلى قاموس البيانات .

4- تحديد متطلبات الوصول إلى الملف وهذا يشـمل المفتـاح الرئيسـي والمفاتيح الثانوية وأسلوب الوصول إلى السجلات وترتيبها ضمن الملف .

5- تحديد الوسيط الذي سيخزن فيه الملف (الأقراص أو الأشرطة الممغنطـة أو غيرها) .

6- تقدير عدد السجلات في الملف انطلاقاً من عدد الكينونـات الحاليـة واحتمالات التوسع في المستقبل .

7- دراسة إمكانية إضافة حقول أخرى يمكن أن تكون مهمة للتعبير عـن النشاط الذي يمثله الملف، مثل تاريخ أخر دفعة للزبون أو للمورد وتـاريخ أخر عملية تمت على السجل وهكذا .

8- حساب طول السجل وحجم الملف وحجم التخزين الـلازم لبيانـات النظام .

9- وضع خطة إنشاء النسخ الاحتياطية للملفات .

2.2.3 تحويل البنية المنطقية للسجلات إلى بنية نظام إدارة قاعدة البيانات:

لقد أشرنا أعلاه أن الطريقة المثلى لتخزين بيانات النظام هي استخدام أنظمة إدارة قواعد البيانات. ويتم تخزين الروابط أو الصلات Links ، الموجودة في البنيـة المنطقية للسجلات ، .في أنظمة إدارة قواعد البيانات بطرق مختلفة . ولذلك فانه قبل

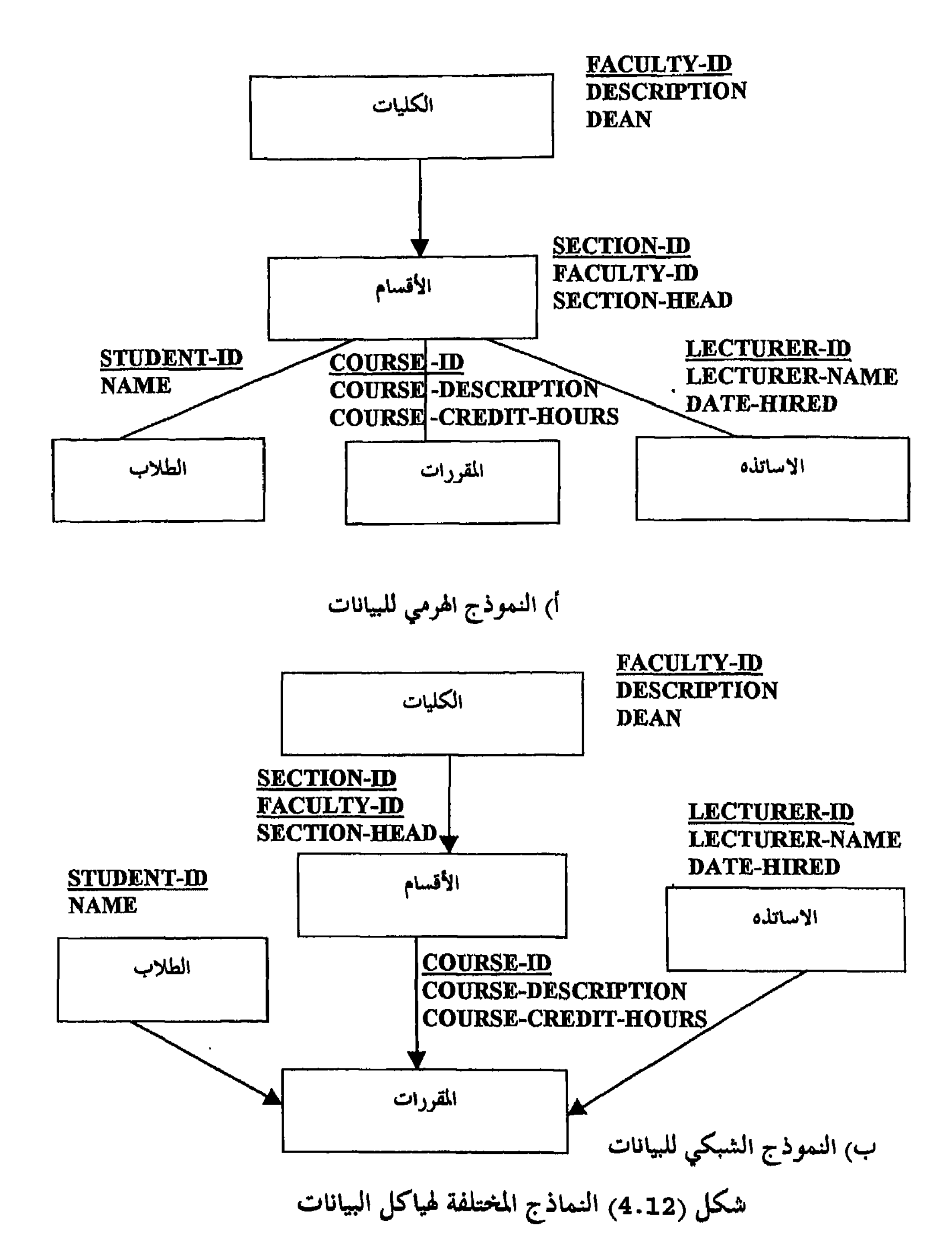
البدء بهذه الخطوة يجب اختيار نظام إدارة قواعد البيانات الذي تناسب بنيته DBMS Structure طبيعة الروابط الموجودة في البنية المنطقية للسجلات.

أ) أنواع الهياكل المستخدمة في أنظمة إدارة قواعد البيانات

بشكل عام فان أنظمة إدارة قواعد البيانات يمكن أن تدعم إحدى الطرق الثلاث التالية لتخزين الروابط بين السجلات :

- الهيكل الهرمي للبيانات Hierarchical Structure .
- الهيكل الشبكي للبيانات Network Structure .
- الهيكل العلاقاتي للبيانات Relational Structure .

وبالرغم من أن الأسلوب الأخير ، أي الهيكل العلاقاتي هو الأبسط والأسهل والأفضل والأكثر استخداماً في قواعد البيانات الحديثة ، إلا انه من المفيد أيضا أن نتعرف على الأساليب الأخرى . ويبين الشكل (4.12) مقارنة بين هذه الأساليب الثلاثة . ونلاحظ من الشكل (4.12 الجزء أ) أنه في النموذج الهرمي للبيانات يتم ربط السجلات بطريقة هرمية حيث يكون لكل سجل أب واحد فقط One Parent . ولذلك يتم تحديد الروابط وتسجيلها في قاعدة البيانات كجزء من السجل المنطقي لذلك السجل . فمثلاً يجب في سجل كل قسم إضافة حقول يتم فيها تسجيل معلومات.الربط Reference links التي تربط سجلات المقررات المختلفة مع الأقسام التي تعتبر هذه المقررات سجلات أبناء لها Child records.وبنفس الطريقة يجب تحديد الصلات التي تربط كل قسم والسجلات التابعة له مثل سجلات الأساتذة وسجلات الطلاب .



أ) النموذج الهرمي للبيانات

ب) النموذج الشبكي للبيانات

شكل (4.12) النماذج المختلفة لهياكل البيانات

الأقسام

رمز القسم	رمز الكلية	الاسم

الكليات

رمز الكلية	الاسم	العميد

الأساتذة

رمز المدرس	الاسم	تاريخ التعيين

المقررات

رمز المقرر	رمز القسم	اسم المقرر	الساعات المعتمدة

الشعب

رمز المقرر	رمز الطلب	الزمن والقاعة

الطلاب

رمز الطالب	اسم الطالب	العنوان

الأعباء التدريسية

رمز المدرس	رمز المقرر	الفصل الدراسي

جـ) النموذج العلاقاتي

تابع شكل (4.12) النماذج المختلفة لهياكل البيانات

وبالرغم من سهولة هذا الأسلوب في تنظيم هيكل قاعدة البيانات إلا انــــه يؤدي إلى تكرارات كثيرة في السجلات المختلفة . كما يتطلب التحديـــد المســبق للصلات أو الروابط وتعريفها أثناء إنشاء القاعدة وقبل إدخال البيانات فيها .

أما الجزء (ب) من الشكل (4.13) فيبين النموذج الشبكي لقاعدة البيانات والذي يسمح أن يكون للسجل الواحد أكثر من أب مما يوفر مرونة أكثر في الوصول إلى السجلات ، وتكرار أقل في البيانات . أما الجزء (جـ) من الشكل (4.12) فيبين النموذج العلاقاتي للبيانات والذي يتكون من مجموعة من الجداول التي تتضمن أقل تكرار ممكن للبيانات . كما أن الصلات أو الروابط Links لا يتم تحديدها بشكل علني Explicit، كما هو الحال في النماذج الهرمية أو الشبكية ، بل بشكل ضمني Implicit من خلال الحقول المشتركة يبين الجداول (العلاقات) المختلفة.

ب) إعداد النموذج المنطقي لقاعدة بيانات النظام

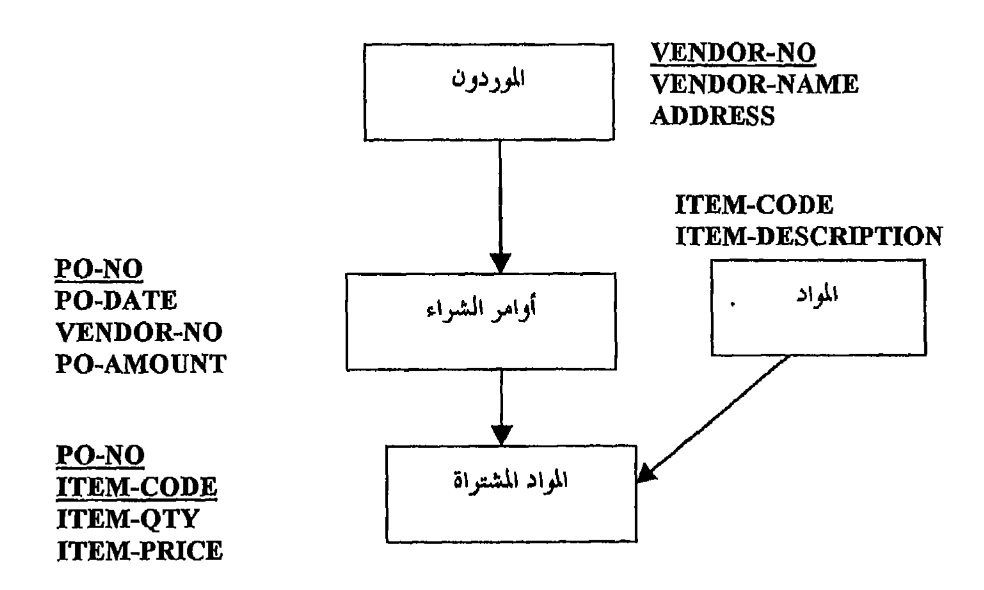
بعد أن تعرفنا إلى البنى المستخدمة في أنظمة إدارة قواعد البيانات، ننتقل إلى دراسة كيفية أعداد النموذج المنطقي لقاعدة بيانات النظام الذي يجري تصميمه. أي إلى الخطوة الثانية من خطوات تصميم قاعدة البيانات، التي يتم من خلالها تحويل البنية المنطقية للسجلات (LRS) إلى نموذج البيانات Data Model المستخدم في نظام إدارة قواعد البيانات الذي تم اختياره لبناء قاعدة بيانات النظام الجديد.

تعتمد طريقة التحويل على نوع نموذج البيانات المستخدم في نظام إدارة قواعد البيانات . ويكون التحويل في غاية السهولة والبساطة عندما يكون نظام إدارة قواعد البيانات الذي سيتم استخدامه علاقاتيا Relational DBMS. ففي هذه الحالة لا نحتاج حتى لإعداد البنية المنطقية للسجلات (LRS)، حيث يتم تحويل العلاقات بشكل مباشر إلى جداول في قاعدة البيانات ، وتكون حقول هذه الجداول عبارة عن خصائص Attributes هذه العلاقات .

أما عندما يكون نظام إدارة قواعد البيانات الذي سيتم استخدامه في النظام الجديد شبكيا Networks DBMS ، فيتم تحويل البنية المنطقية للسجلات إلى نموذج شبكي على النحو التالي :

- تحويل السجلات Record Types الموجودة في البنية المنطقية للسجلات (LRS) إلى سجلات البنية الشبكية.

- تحويل الروابط بين أنواع السجلات المختلفة إلى روابط تصل بين سجلات الشبكة كما هو مبين في الشكل (5.12) .



شكل (5.12) البنية الشبكية لقاعدة البيانات

وأخيرا في حال استخدام نظام إدارة قواعد بيانات يستخدم النماذج الهرمية، فإن عملية التحويل تتطلب الانتباه إلى عامل أخر وهو عدم إمكانية ربط النوع

الواحد من السجلات إلا بنوع أخر واحد فقط هو سجل الأب لذلك السجل. وهذا قد يتطلب من المصمم المقارنة بين إمكانات الربط المختلفة لتحديد السجلات الأب Parent Records والسجلات الأبناء Child Records والمعيار هنا اختيار الطريقة التي تساعد على تقليل التكرار في البيانات Data Redundancy إلى أدنى حد ممكن. وهكذا فانه في نهاية هذه الخطوة نكون قد توصلنا إلى النموذج المنطقي للبيانات والذي يمكن أن يكون شبكيا أو هرميا أو علاقاتيا .

4. التصميم المادي لقاعدة البيانات Physical Database Design :

بعد الانتهاء من إعداد النموذج المنطقي لقاعدة البيانات، والذي يتناسب مع البنية المعتمدة في نظام إدارة قواعد البيانات المستخدم لبناء بيانات النظام ، ننتقل في هذه الخطوة إلى إعداد مواصفات التصميم المادي لقاعدة بيانات النظام والتي تتضمن ما يلي :

- **تحديد بنية الملفات المادية والفهارس** Indexes التي سيتم على أساسها ترتيب السجلات في الملفات المادية . وذلك في ضوء متطلبات الوصول التي تم تحديدها عند إعداد توصيف قاعدة البيانات .
- **تحسين أداء عمليات الوصول إلى البيانات** : وترتبط هذه العمليات عادة بمسارات الوصول Access Paths التي يتم من خلالها تحديد طرق التنقل بين السجلات للوصول إلى المعلومات المطلوبة . وقد يتطلب الأمر استبدال مسارات الوصول الطويلة بمسارات اقصر، والاضطرار إلى تكرار بعض حقول البيانات لتقصير هذه المسارات . وهنا يمكن للمحلل المفاضلة بين

الأداء السريع ومتطلبات التمثيل الجيد للبيانات ، واختيار الحلول المناسبة لهذه الحالات .

- **تحديد الإجراءات اللازمة لحماية قاعدة البيانات** ومنع تخريب البيانات في حال وقوع كارثة أو حدوث خطأ ما . ولمواجهة هذه الحالات التي يمكن أن تقع عن قصد أو بدون قصد ، يجب وضع إجراءات تضمن حماية قاعدة البيانات من خلال تلافي وقوعها أو تقليل تأثيرها إلى أدنى حد ممكن. ومن هذه الإجراءات أخذ **النسخ الاحتياطية** Backup لقاعدة البيانات بأكملها وبشكل دوري، كما يمكن اخذ نسخ احتياطية بشكل يومي للأجزاء الهامة فيها. وفي حال حدوث أخطاء أو أعطال تؤدي إلى فقدان قاعدة البيانات، يمكن استخدام هذه النسخ الاحتياطية لاستعادة قاعدة بيانات النظام Database Recovery، علماً أن العديد من أنظمة إدارة قواعد البيانات تقوم بمسك ملفات خاصة Journal files يتم فيها تسجيل جميع العمليات والمهام التي يتم تنفيذها في قاعدة البيانات. وفي حال حدوث أخطاء أو أية أعطال يمكن استخدام هذه الملفات لاستعادة قاعدة البيانات بشكل سريع وفوري.

أسئلة وتمارين الفصل

1- اشرح أهداف ومبادىء ومراحل تصميم قاعدة البيانات ؟

2- ما المقصود بالتحليل العلاقاتي وما هي خطواته ؟

3- اشرح خطوات تطبيع البيانات باستخدام مثالا على ذلك ؟

4- اشرح الخطوات الرئيسية الثلاث لمرحلة تصميم قاعدة البيانات ؟

5- ما المقصود بالبنية المنطقية للسجلات ومم تتكون ؟

6- اشرح كيفية إعداد البنية المنطقية للسجلات انطلاقا من النموذج العلاقاتي لبيانات النظام ؟

7- اشرح كيفية إعداد البنية المنطقية للسجلات انطلاقا من مخطط الكينونة - العلاقة ؟

8- ما هي الأنواع المختلفة للملفات المستخدمة في نظم المعلومات ؟

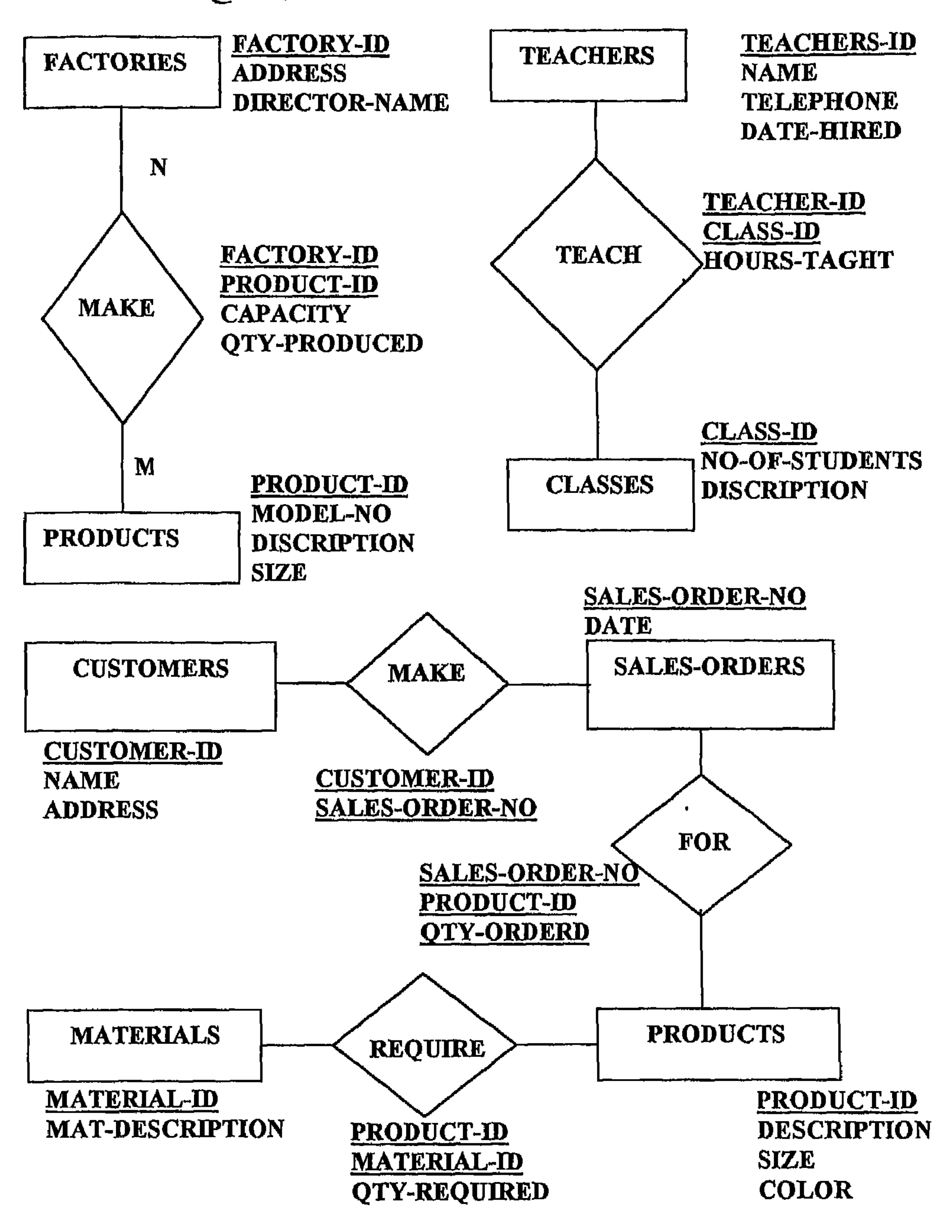
9- اشرح خطوات تصميم الملفات ؟

1.- قارن بين البنى الهرمية والشبكية والعلاقاتية لقواعد البيانات ؟

11- ما هي المهام التي يتضمنها التصميم المادي لقاعدة البيانات ؟

تمارين

1- أكتب مخططات الكينونة - العلاقة التالية بشكل نماذج علاقاتية :



2- ادرس الحالة التالية وصمم مجموعات العلاقات الطبيعية للبيانات الـــتي تمثلها ؟

يعرف المهندسون العاملون في الشركة من خلال رموزهـــم PERSON-ID وأسمائهم NAMES، ويخصص لهم أرقام صلاحيـــة AUTHORITY-NO لكل مشروع من مشاريع الشركة . ويمكن أن يعطي المهندس الواحد أكثر من رقم صلاحية . كما أن عدة مهندسين يمكن أن يعطوا أرقام صلاحيـــة للمشروع الواحد .

تعرف مشروعات الشركة من خلال رموزها PROJECT-ID ومدرائـــها MANAGER والميزانية المخصصة لها BUDGET.

تعرف صلاحيات المهندسين من خلال رقم الصلاحيـــة وتـــاريخ منحـــها AUTHORITY-DATE والحد الأقصى للمبلغ الذي يمكن صرفـــه -MAX AMOUNT لكل مشروع من المشاريع .

يقوم المهندسون بإصدار طلبات الشراء لتأمين المواد اللازمـــة للمشـــروع حسب الصلاحيات الممنوحة لهم . يتألف الطلب من الرقـــم ORDER-NO والتاريخ ORDER-DATE ورقم الصلاحية وعدد غير محدد مـــن المـــواد (أسمائها MATERIAL-NAME وكمياتها QTY-REQUIRED والســـعر PRICE والقيمة AMOUNT) .

يوجه الطلب لمورد واحد ويعرف المـــوردون برموزهـــم VENDOR-ID وأسمائهم VENDOR-NAME وعناوينهم VENDOR-ADDRESS .

يرسل الموردون فواتيرهم التي تتضمن البيانات التاليـــة : رقـــم الفـــاتورة INVOICE-NO وتاريخها INVOICE-DATE ورقم الطلب ORDER-NO

وتفاصيل المواد المرسلة (أسمائها وكمياتها وأسعارها) والمبلـــغ الإجمـــالي للفاتورة TOTAL-AMOUNT والتاريخ الذي يجب دفع الفاتورة خلالـــه للحصول على حسم لقاء الدفـــع المبكـــر EARLY-PAYMENT-DATE ونسبة هذا الحسم EARLY-PAYMENT-RATE .

3- ادرس الجدول المبين أدناه وأجب على الأسئلة التالية :

-اكتب العلاقة التي تمثل هذا الجدول وحدد مفاتيحها ؟

--حول هذه العلاقة إلى الشكل الطبيعي الثالث ؟

الامتحانات :

اسم المقرر	اسم المحاضر	اسم الطالب	تاريخ الامتحان	رقم القاعة
برمجة	د. علي سعيد	علي حسن	1998/6/15	1240
برمجه	د. علي سعيد	اياد اسماعيل	1998/6/15	1240
بحوث عمليات	د. محمد حسن	محسن خيري	1998/6/17	1211
تحليل أنظمة	د. احمد الكاتب	علي حسن	1998/6/18	1340
بحوث عمليات	د. ابراهيم محمود	اسماعيل زعتر	1998/6/17	1211
تحليل أنظمة	د. احمد الكاتب	محسن خيري	1998/6/18	1340
تقنية المعلومات	د. هيثم بعلي	علي حسن	1998/6/25	1312
أتمتة مكاتب	د. شادي عمر	اياد اسماعيل	1998/6/20	1217
أتمتة مكاتب	د. شادي عمر	محسن خيري	1998/6/20	1217

الفصل الثالث عشر

تصميم برامج النظام

إن تصميم برامج النظام هو أحد الأنشطة الرئيسة لمرحلة التصميم التفصيلي للنظام الجديد. وتبدأ عملية تصميم البرامج انطلاقا من النموذج المادي للنظام الجديد، الذي تم التوصل إليه خلال مرحلة التصميم العام، وبشكل خاص من مخططات تدفق البيانات في النظام الجديد، والتي تم فيها تحديد مناطق الحوسبة Computerization areas. فالعمليات الموجودة في هذه المناطق (العمليات التي سيتم حوسبتها في النظام الجديد) تمثل نقطة البدء في إعداد التصاميم المتعلقة بالبرامج اللازمة لبناء النظام الجديد. وسندرس في هذا الفصل الخطوات التي يتم خلالها تصميم برامج النظام، ثم سنتعرف بشكل تفصيلي على واحدة من أهم أدوات تصميم النظم وهي المخططات الهيكلية Structure Charts وسندرس كيفية إعداد هذه المخططات واستخدامها في تصميم البرامج.

إن بناء نظام معلومات محوسب يحتاج، بشكل عام إلى ثلاثة أنواع من البرامج هي:

- برامج بناء وتحديث قاعدة بيانات النظام.
- برامج إسترجاع البيانات من القاعدة.
- برامج إعداد التقارير المختلفة.

ونظراً لأهمية استرجاع البيانات وإعداد التقارير ومتطلبات الاستجابة السريعة لاحتياجات المستخدمين المتغيرة، فقد تم تطوير لغات برمجة حديثة تسهل

عمليات الاسترجاع وبناء الاستعلامات المختلفة. وتتضمن لغات الجيل الرابع العديد من الإمكانات والتسهيلات في هذا المجال مثل لغات الاستعلام البنيوية (SQL) Structured Query Languages ومولدات التقارير Report Generators وغيرها، ولقد أتاحت هذه اللغات إمكانية الحصول على التقارير والمعلومات اللازمة دون الحاجة إلى الرجوع للمبرمجين، فيكفي أن يحدد المستخدم المعلومات اللازمة له، والأسلوب الذي يفضله لعرض هذه المعلومات، ويترك بقية التفاصيل المتعلقة بالإجراءات اللازمة لإيجاد هذه البيانات ومعالجتها لتتم بصورة تلقائية من قبل النظام.

أما بالنسبة للبرمجيات اللازمة لبناء وإدامة قواعد البيانات وتحديثها فيتم كتابتها بلغات البرمجة Programming Languages التي يمكن أن تكون من اللغات الموجهة نحو الإجراءات Procedure - Oriented أو من لغات الجيل الرابع، ومن الطبيعي أن هذا النوع من البرامج مهم جداً لضمان دقة وحداثة وسلامة البيانات الموجودة في القاعدة.

1. خصائص البرامج الجيدة.

بشكل عام، يجب أن تتصف البرامج الحاسوبية بما يلي:

- **تلبية متطلبات المستخدم:** أي تقوم بالوظائف المطلوبة منها بكفاءة وفاعلية.
- **سهولة القراءة والفهم:** أي يمكن فهمها بسرعة وسهولة، وذلك لتمكين استخدامها وتطويرها من قبل جميع المشاركين في عملية البرمجة.
- **سهولة الصيانة:** أي يمكن إدخال التغيرات والتعديلات فيها بسهولة.

- **الكفاءة في استخدام الموارد:** أي تتضمن الاستخدام الأمثـــل لجميـــع موارد نظام المعلومات من تجهيزات وبرمجيات وغيرها.

ولإعداد برامج النظام وفق القواعد المشار إليها أعلاه يجب اســـتخدام الأســـلوبين التاليين:

- تصميم البرامج بشكل وحدات وظيفية Modular Program Design
- البرمجة الهيكلية للوحدات الوظيفية Structured Programming

1.1 تصميم البرامج بشكل وحدات وظيفية:

وفقاً لهذا الأسلوب، يتم تقسيم النظام إلى وظائف محددة بشـــكل جيـــد Well-Defined Functions ثم يتم تخصيص كل وظيفة (برمجة كل وظيفـــة) ضمـــن وحدة وظيفية واحدة تسمى Program Module.

تتم عملية تقسيم النظام إلى وظائفه المكونة لـــه، منـــذ بدايـــة عمليـــات التحليـــل، أثناء إعداد مخططات التدفق المختلفـــة (DFDs). ثم تســـتخدم هـــذه المخططات لبناء النموذج المادي للنظام الجديد، الذي تكون مخططات التدفق أحـــد مكوناته الرئيسة. وتمثل هذه المخططات بجميع مستوياتها التقسيم الأمثل لعمليـــات النظام، والتي سيتم برمجتها (أي العمليات) في وحدات وظيفية ترتبط مـــع بعضـــها البعض بعلاقات هرمية (أي ضمن مستويات مختلفة). وهذا ما سندرســـه في هـــذا الفصل عند إعداد المخططات الهيكلية لبرامج النظام.

2.1 البرمجة الهيكلية Structured Programming

يستخدم هذا الأسلوب لكتابة البرامج الخاصة بـــالوحدات الوظيفيـــة، أي في مرحلة التنفيذ، أثناء عمليات البرمجة والاختبار. وتتضمن البرمجة الهيكلية استخدام ما

يسمى بـــ بنى التحكم Control Structures لتحسين منطق البرنامج وجعله أكــــثر وضوحاً، وبالتالي تسهيل تعديله وصيانته، وتساعد بنى التحكم هـــــذه في تطبيـــق أسلوب تطوير البرامج من الأعلى إلى الأسفلTop-Down Programming. حيـــث يمكن البدء بكتابة واختبار البرنامج الرئيسي للنظام ثم الانتقال إلى كتابـــة الـــبرامج التابعة له، ثم إلى البرامج الفرعية التابعة لها، ثم إلى البرامج الفرعية التابعة لكل برنامج وهكذا. أما بنى التحكم التي تستخدم في البرمجة الهيكلية فهي :

- بنية التسلسل Sequence Structure.
- بنية القرار Decision Structure.
- بنية التكرار Repeat Structure.

وهكذا يجب تقسيم البرنامج ضمن الوحدة الوظيفية إلى مقــــاطع Blocks يتضمن كل منها مجموعة من التعليمات، ويتم استخدام هذه البنى المذكورة أعـــلاه لكتابة هذه المقاطع والربط فيما بينها.

ولكي يكون برنامج الوحدة الوظيفية اكثر وضوحاً يجـــب تســـمية هـــذه المقاطع، ويمكن كتابة الملاحظات التي تشرح وظيفتها ضمن البرنامج، بالإضافــة إلى الإشارة إلى وثائق النظام الأخرى التي يمكن أن تساعد في فهم البرنـــامج، وتجـــدر الإشارة هنا إلى أهمية كتابة توصيف العمليات Process Specification بشكل بنيوي باستخدام لغات التوصيف البنيوية مما يساعد كثيراً في تطبيق هذا الأسلوب.

2. خطوات تصميم برامج النظام

يتم تصميم البرامج اللازمة لنظام المعلومات الذي يجـــري تطويـــره وفقـــاً للخطوات الرئيسة التالية:

1.2 تحديد الأنظمة الفرعية المطلوب حوسبتها:

في هذه الخطوة يتم تجميع العمليـــات الموجـــودة في منـــاطق الحوســـبة Computerization Areas في مخططات تدفق بيانـــات النظـــام الجديـــد، بشـــكل مجموعات يطلق عليها الأنظمة الفرعية للحوسبة Computer subsystems، وتتـــألف هذه الأنظمة الفرعية من مجموعة من العمليات المرتبطة منطقياً مع بعضها البعـــض، مثلاً مجموعة عمليات متتابعة لأداء وظيفة معينة (قراءة وتدقيق وإدخـــال بيانـــات الفاتورة مثلاً)، أو مجموعة من عمليات تتعامل مع مخزن بيانـــات واحـــد (إضافـــة سجلات أو تعديلها أو حذفها). وغالباً يتم التركيز في هذه الخطوة علـــى تجميـــع العمليات المطلوب حوسبتها ضمن مجموعات ليتم تصميم برامج النظام وفق هيكلية هرمية تبدأ بالبرنامج الرئيس للنظام الذي يتم من خلاله استدعاء برامـــج الأنظمـــة الفرعية المختلفة، والتي تقوم بدورها باســـتدعاء برامـــج العمليـــات (الوحـــدات الوظيفيـــة) التابعة لها، وهكذا. فمثلاً في نظام المبيعات يمكن تجميع العمليات المتعلقة باستلام الطلبات من الزبائن وتدقيقها والتأكد من توفر المنتجات المطلوبة وإصـــدار أمر التسليم للمخازن لإرسالها إلى الزبائن في نظام فرعي واحد هو نظـــام معالجـــة الطلبات، كما يمكن تجميع العمليات المتعلقة بتنفيذ هذه الطلبات أي تجهيز المنتجات المطلوبة وتغليفها وشحنها إلى الزبائن في نظام فرعي آخر وهو نظام توريد أو توزيع الطلبات إلى الزبائن.

2.2 التوصيف التفصيلي للعمليات Detailed Process Specification

بعد تجميع العمليات ضمن أنظمة فرعية للحوسبة يجب إعداد التوصيفـــات التفصيلية لكل عملية من هذه العمليات التي سيتم حوسبتها. تمهيداً لتحويل هـــذه

العمليات إلى وحدات وظيفية Program Modules تقوم كل منها بأداء عملية واحدة من هذه العمليات. ويتم هذا التوصيف استناداً إلى توصيف عمليات النظام التي تم إعدادها خلال المراحل السابقة.

3.2 إعداد المخطط الهيكلي للنظام Structure Chart

يتم في هذه الخطوة إعداد المخطط الهيكلي الذي يربط الوحدات الوظيفية للنظام وفق العلاقات الهرمية لها. والمخطط الهيكلي هو عبارة عن تمثيل هرمي لمخططات تدفق البيانات ذات الطبيعة الشبكية. وسندرس في هذا الفصل بشكل تفصيلي كيفية إعداد هذه المخططات الهيكلية.

4.2 التوصيف التفصيلي للوحدات الوظيفية

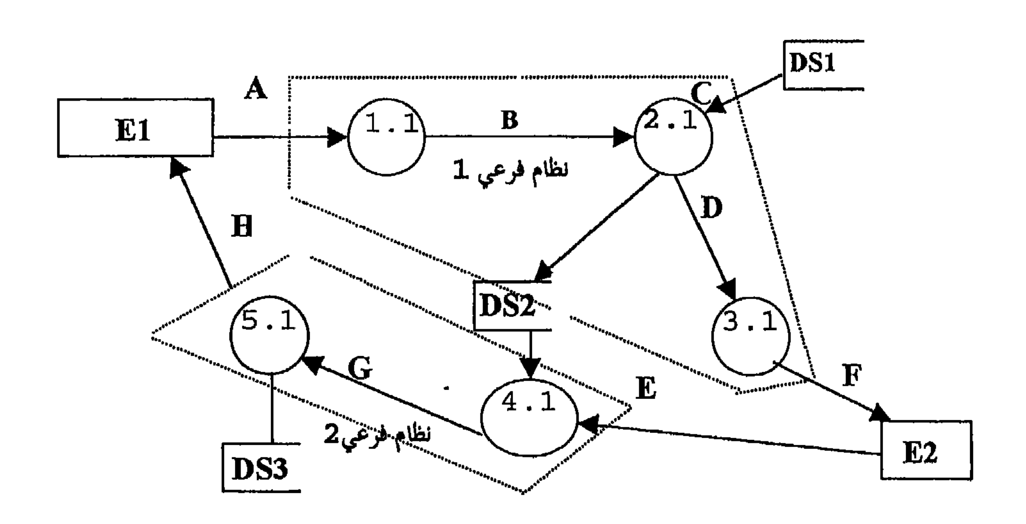
بعد رسم المخطط الهيكلي لبرامج النظام، يتم في هذه الخطوة إعداد التوصيفات التفصيلية للوحدات الوظيفية المكونة لهذه المخططات، ويحتوي هذا التوصيف عادة على المعلومات التالية:

- **المدخلات** : أي البيانات التي يجب إرسالها إلى الوحدة الوظيفية عند استدعائها من قبل البرنامج (الوحدة الوظيفية) الذي تتبع له هذه الوحدة.
- **المعالجة**: أي توصيف الخوارزمية اللازمة لتنفيذ العمليات المطلوبة لمعالجة البيانات الداخلة وتحويلها إلى المخرجات المطلوبة.
- **المخرجات**:.أي المعلومات التي سيتم إنتاجها من قبل هذه الوحدة الوظيفية وتمريرها إلى العمليات الأخرى في النظام.

5.2 تصميم الحزم Package Design

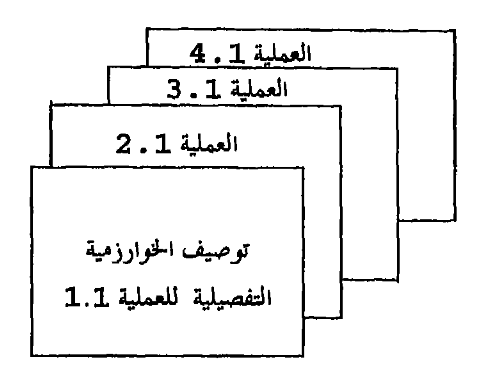
بعد الانتهاء من إعداد التوصيفات التفصيلية للوحدات الوظيفية، يتم ربط الوحدات التنفيذية مع الوحدات في المستوى الأعلى لها، ويطلق على المجموعات الناتجة وحدات التحميل Load Modules، وتسمى عملية التجميع هذه بالتحزيم Packaging أو تصميم الحزم، والحزمة كما هو واضح يمكن أن تتكون من وحدة وظيفية واحدة في مستوى معين، مع وحدة أو عدة وحدات في المستوى الأدنى، ويشرح الشكل (1.13) الخطوات الخمس لتصميم البرمجيات.

أ) الخطوة الأولى : تحديد الأنظمة الفرعية المطلوب حوسبتها

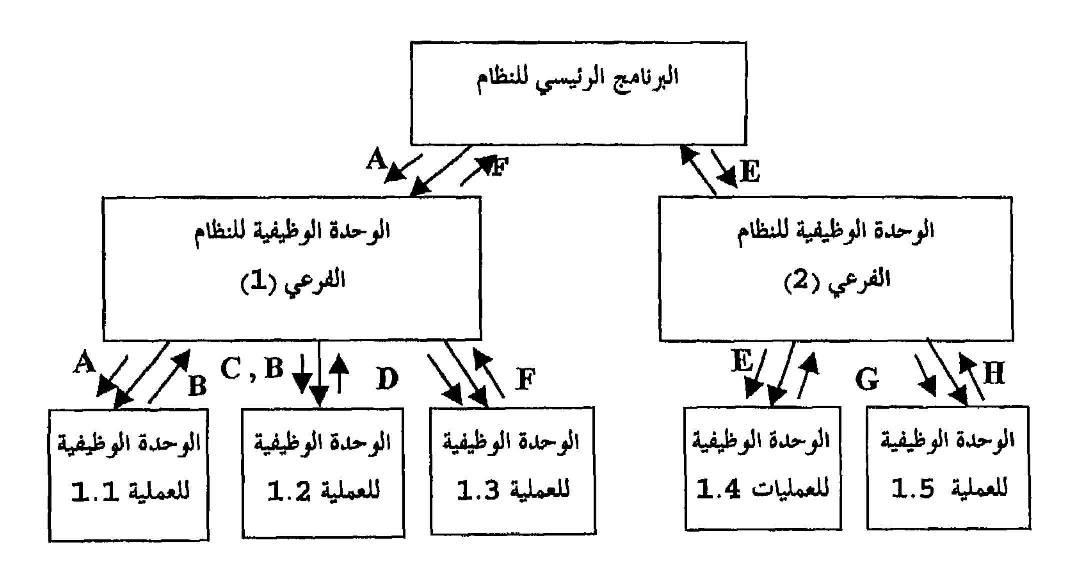


شكل (1.13) الخطوات الخمس لتصميم البرمجيات

ب) الخطوة الثانية : التوصيف التفصيلي للعمليات المطلوب حوسبتها

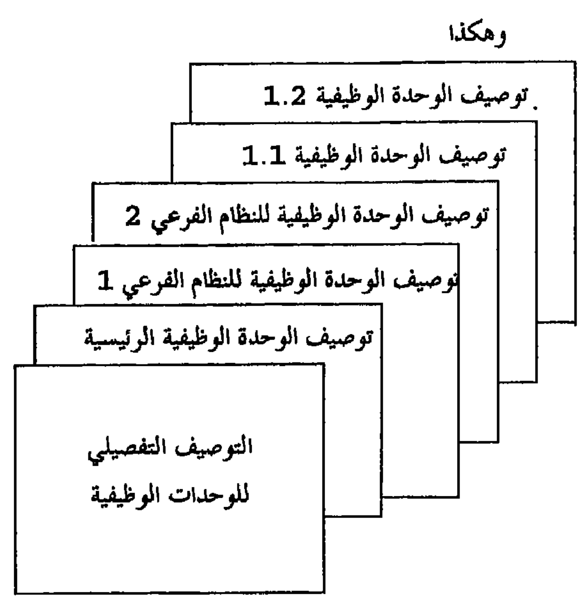


جـ) الخطوة الثالثة : إعداد المخطط الهيكلي للنظام

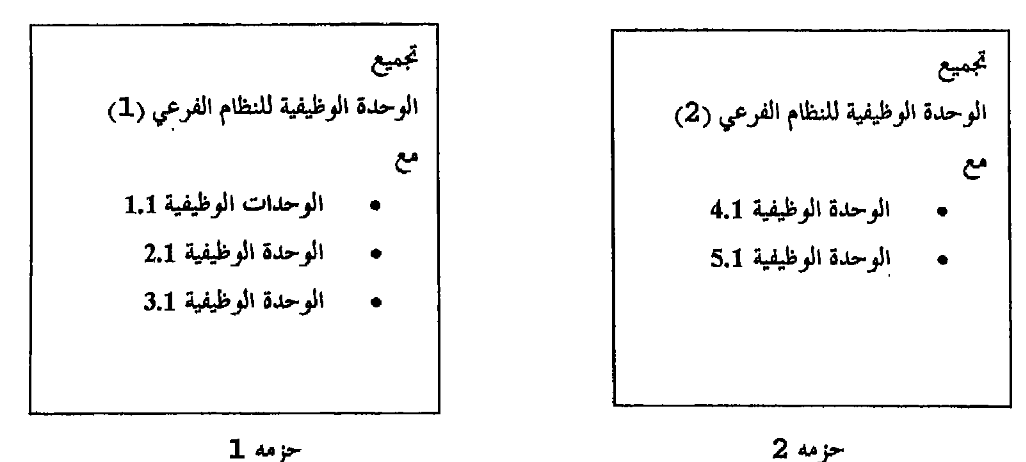


تابع شكل (1.13) الخطوات الخمس لتصميم البرمجيات

حـ) الخطوة الرابعة : التوصيف التفصيلي للوحدات الوظيفية



د) الخطوة الخامسة : تصميم الحزم



تابع شكل (1.13) الخطوات الخمسة لتصميم البرمجيات

3. المخططات الهيكلية Structure Charts ومكوناتها:

لقد تعرفنا من خلال خطوات عملية تصميم البرمجيات إلى أنها تتضمن تحويل مخططات تدفق البيانات إلى مخطط هيكلي لبرامج النظام. والمخططات الهيكلية هي عبارة عن أداة بيانية تستخدم لتمثيل العلاقة الهرمية التي تربط الوحدات الوظيفية للنظام مع بعضها البعض.

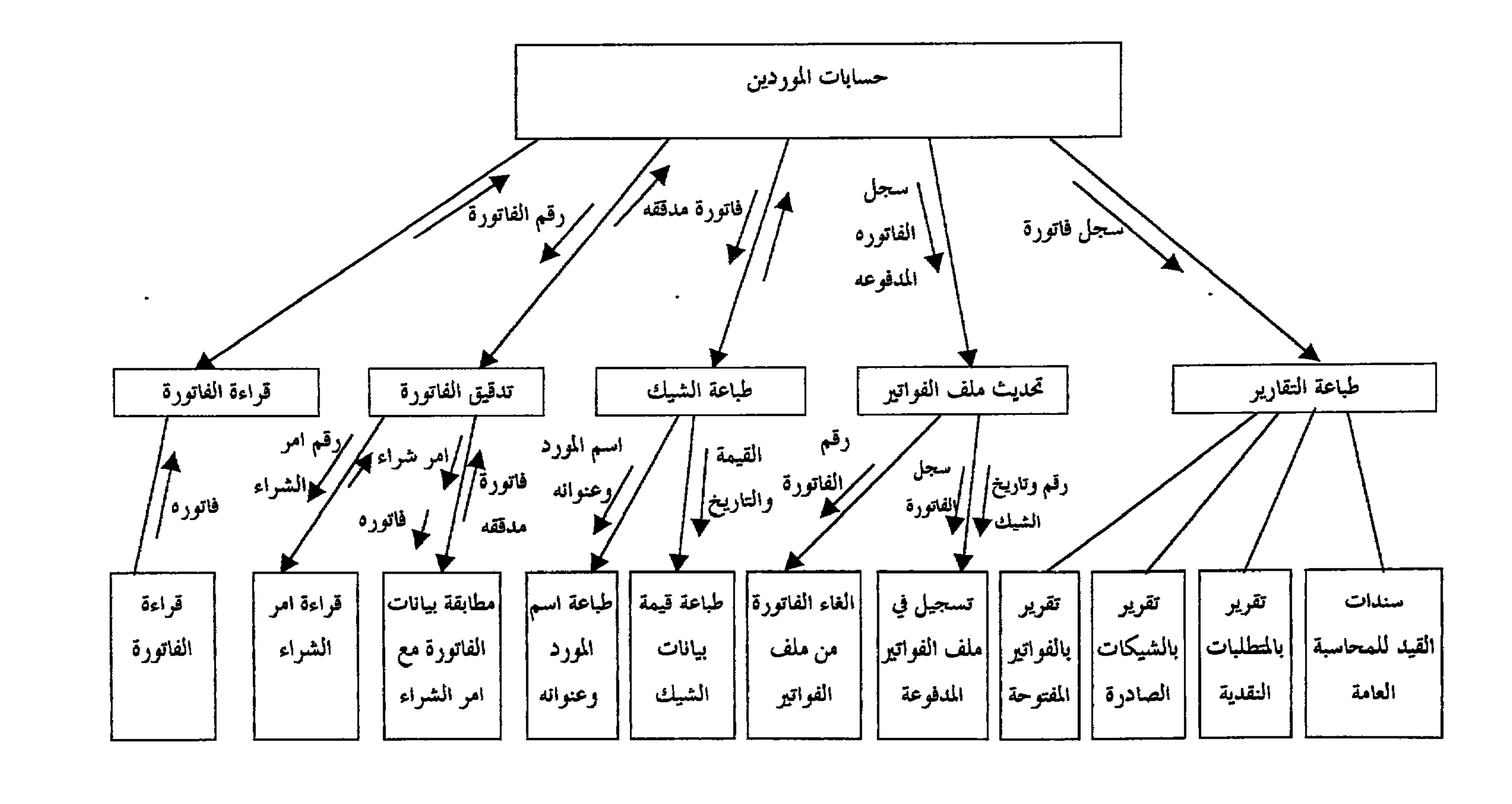
تتكون المخططات الهيكلية من **العناصر الأربعة التالية** والموضحة في الشكلين (2.13) و (3.13).

1.3 الوحدات الوظيفية : Program Modules

يرمز للوحدات الوظيفية في المخطط بشكل مستطيل، وتكتب تسميتها في داخله. والوحدة الوظيفية هي عبارة عن مجموعة من التعليمات أو التصريحات Instructions and Statements التي تؤدي مهمة أو عملية محددة، وغالباً تكافئ الوحدة الوظيفية عملية واحدة في مخططات تدفق البيانات، أما التجسيد المادي لهذه الوحدة فيمكن أن يكون بشكل برنامج فرعي Subroutine يتم استدعاءه خلال عمليات المعالجة.

تتواجد الوحدات الوظيفية في المخططات الهيكلية في عدد من المستويات، ويبدأ المخطط من القمة بوحدة وظيفية واحدة تسمى الوحدة الرئيسة Main Module، يليها في المستوى التالي مباشرة عدد من الوحدات التابعة Subordinate Modules يتم استدعائها من قبل الوحدة الوظيفية الرئيسة، وتقوم الوحدات الوظيفية التابعة بدورها باستدعاء الوحدات الوظيفية التابعة لها والموجودة في مستوى أدنى وهكذا. وبشكل عام يستخدم مصطلح الوحدة المشرفة Supervisor للدلالة على

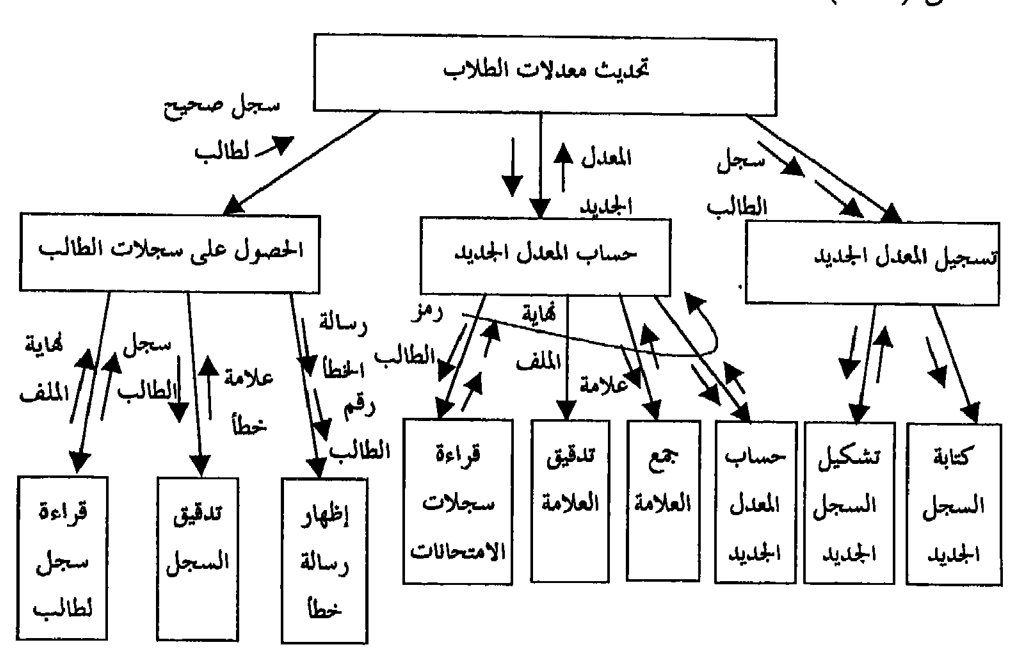
/



الشكل (2.13) مخطط هيكلي (مثال لنظام حسابات الموردين)

الوحدة التي تقوم بالاستدعاء، والوحدة التابعة Subordinate للدلالة على الوحدات التي يتم استدعاءها من قبل وحدة وظيفية أعلى منها.

وكما هو الحال في المخططات البيانية عموما فان الاتجاهات تكـون مـن الأعلى للأسفل ومن اليسار نحو اليمين. ولذلك يتم رسم الوحدات الوظيفية الـتي تقوم باستقبال المدخلات ومعالجتها في أقصى يسار المخطط الهيكلي، أما الوحـدات الوظيفية التي تقوم بإنتاج المخرجات المطلوبة من النظام فتوضع في أقصى اليمـين، وهكذا فان الوحدات الوظيفية لعمليات التحويل أو المعالجة تكون دوما في وسـط المخطط. وسريان عمليات المعالجة ضمن المخطط الهيكلي يكون من اليسار باتجـاه اليمين على النحو التالي: مدخلات ثم معالجة ثم مخرجات، على النحـو المبـين في الشكل (3.13).



شكل (3.13) المخطط الهيكلي لنظام تحديث معدلات الطلاب

2.3 الروابط Connections

تستخدم الروابط لتمثيل العلاقة بين الوحدات الوظيفية، وترسم غالبـــاً في المخطط بشكل سهم ينطلق من الوحدة المشرفة إلى الوحدة التابعة (مـــن الوحـــدة المستدعية Calling Module إلى الوحدة المستدعاة Called Module)، وتـــدل هـــذه الروابط ضمناً على العودة غير المشروطة إلى الوحدة المشرفة، بعد الانتهاء من تنفيـــذ برنامج الوحدة الوظيفية التابعة.

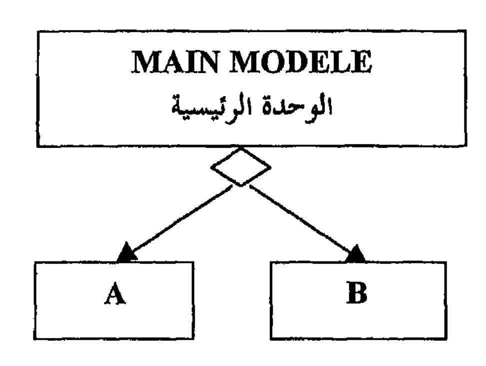
3.3 الثنائيات أو المزدوجات Couples

عندما تقوم الوحدة الوظيفية الرئيسة أو المشرفة باستدعاء وحدة وظيفيـــة تابعه لها، فإنها قد تمرر بعض فقرات من البيانات اللازمـــة لقيـــام هـــذه الوحـــدة بالعمليات المطلوبة منها، كما أنه بعد انتهاء عمل الوحدة المســـتدعاة، قـــد تقـــوم بدورها بتمرير بيانات (إعادة بيانات) إلى الوحدة المشرفة. ويتم تمثيل هذه البيانـــات التي يتم تمريرها من الوحدة المشرفة إلى الوحدة التابعة وبـــالعكس بالمزدوجـــات أو الثنائيات Couples، وترسم بشكل اسهم صغيرة بمحاذاة الروابط الـــتي يتـــم مـــن خلالهـــا تمرير هذه البيانات ، كما هو مبين في الشكل (2.13) كما يمكن أن تمثـــل هذه الثنائيات إشارات تحكم تدل على حالة أو ظرف معين مثل نهايـــة الملـــف أو وجود خطأ ما في السجل أو غير ذلك.

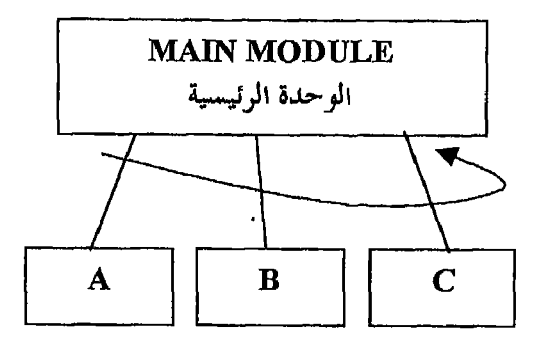
4.3 رموز تسلسل التنفيذ Execution Sequence

لقد ذكرنا سابقاً أن تسلسل تنفيذ الوحدات الوظيفية الموجودة في المخططـــات الهيكلية يكون بشكل عام من اليسار نحو اليمين ولكنه في الحالات التي تتطلب تغـــير هذا التسلسل،فانه يمكن استخدام الرموز التالية والمبينة في الشكل (4.13):

- رمز المعين (القرار) : ويرسم في أسفل الوحدة المشرفة ويدل على أنه يتـــم تنفيذ وحدة وظيفية واحدة فقط من الوحدات المرتبطة به.
- رمز التكرار: وهو بشكل سهم دائري يدل على أن الوحدات المشمولة بـه يتم استدعائها بشكل متسلسل واحدة بعد أخرى وبشكل متكرر.



أ) رمز القرار : يتم استدعاء إحدى الوحدتين الوظيفيتين فقط (A أو B)



ب) رمز التكرار : يتم استدعاء الوحدات الوظيفية A ثم B ثم C بشكل متكرر

شكل (4.13) رموز تسلسل التنفيذ في المخططات الهيكلية

4. تراكيب المخططات الهيكلية Structure Charts Constructs

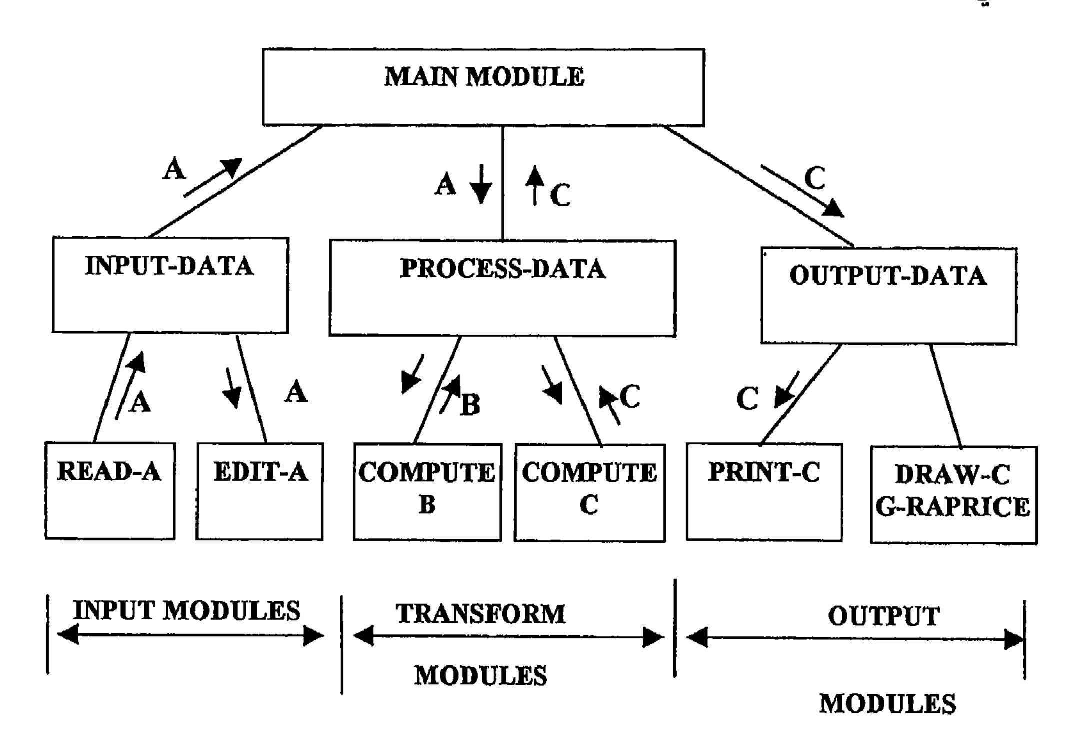
يتم إعداد المخططات الهيكلية بشكل عام وفق أحد التركيبتين التاليين:

- التركيبة التحويلية Transform-Centered Construct
- تركيبة العمليات Transaction-Centered Construct

وبالرغم من أن معظم المخططات الهيكلية يمكن أن تحتوى هاتين الـتركيبتين في نفس الوقت إلا اننا سندرسهما بشكل منفصل كل على حدة.

1.4 تركيبة التحويل: Transform - Centered Construct

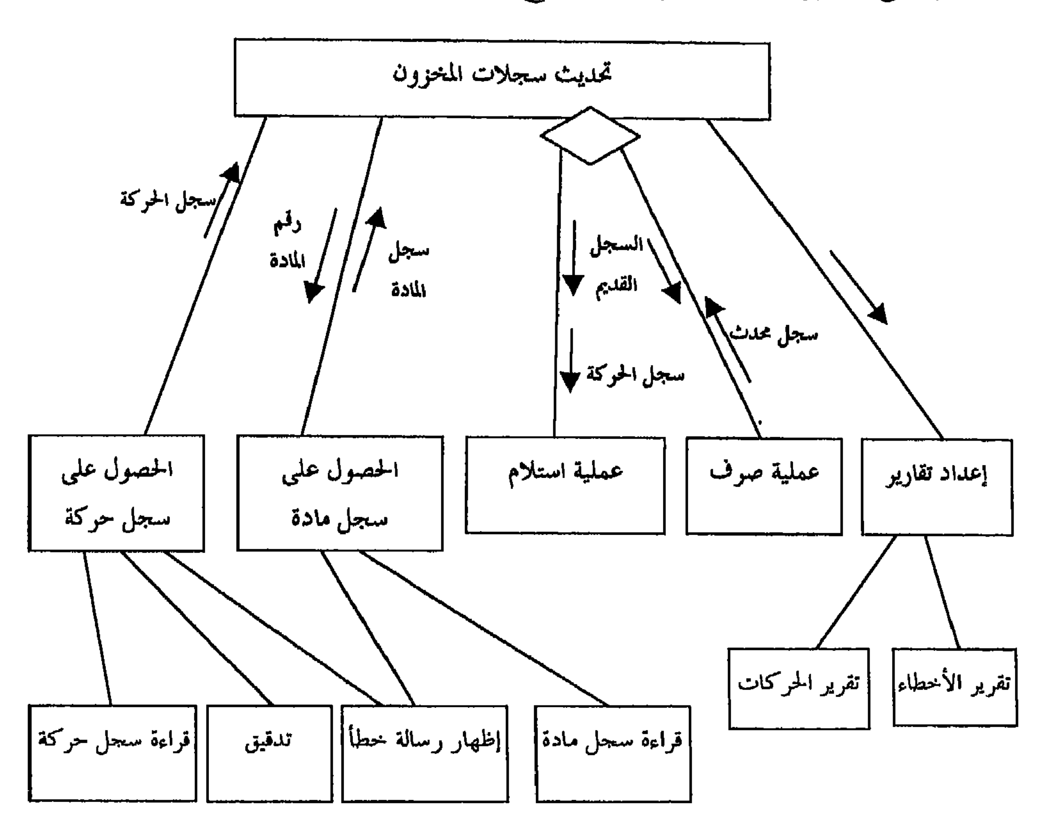
تقوم هذه التركيبة للمخططات الهيكلية باستخدام المدخـــلات وتدقيقـــها وتحويلها إلى المخرجات المطلوبة من خلال سلسلة من العمليات، حيث تكون كـــل عملية بشكل وحدة وظيفية Module. وفقا لهذه التركيبة يمكن تقسيم الوحـــــدات الوظيفية للنظام إلى ثلاثة أنواع هي: الوحدات الوظيفيـــة للإدخـــال والوحـــدات الوظيفية للمعالجة والوحدات الوظيفية للإخراج كما هو مبين في الشـــكل (5.13) التالي:



شكل (5.13) مخطط هيكلي ذو تركيبة تحويلية

2.4 تركيبة العمليات Transaction - Centered Construct

تستخدم هذه التركيبة لإعداد المخططات الهيكلية لبرامج الأنظمة التي تقوم بمعالجة أنواعا مختلفة من العمليات. وفي هذه المخططات تقوم الوحدة الوظيفية الرئيسة باستدعاء إحدى الوحدات التابعة لها في ضوء العملية المطلوبة، كما هو مبين في الشكل (6.13)، حيث تقوم الوحدة الرئيسية في كل مرة باستدعاء الوحدة الوظيفية لمعالجة عمليات الاستلام أو الوحدة الوظيفية الخاصة بمعالجة عمليات السحب من المخزون، وذلك في ضوء نوع العملية.



شكل (6.13) مخطط هيكلي (تركيبة العمليات)

5. خطوات إعداد المخططات الهيكلية:

يتم إعداد المخططات الهيكلية بالاستناد إلى مخططات تدفـق البيانـات في النظام الجديد، فهذه المخططات تحتوي على العمليات التي يجب أن يقوم بها النظام. تمثل المخططات الهيكلية بشكل عام الوحدات الوظيفية للنظام، أي البرامج التي سيتم من خلالها تنفيذ كل عملية من عمليات النظام. وهكذا فان كل عملية من العمليات الموجودة في مخططات التدفق سيتم تحويلها إلى وحدة وظيفية في المخطط الهيكلـي Structure Chart . .

وهكذا فان عملية إعداد المخططات الهيكلية هي عبارة عن عمليـة تحويـل لمخططات تدفق البيانات ذات الطبيعة الشبكية، وإعادة رسمها بشكل مخطط هرمـي يتم فيه تمثيل العلاقات الهرمية بين العمليـات (أي الوحـدات الوظيفيـة). أمـا خطـوات هذه العملية فهي على النحو التالي:

1) إنشاء الوحدة الوظيفية الرئيسة والوحدات التابعة لها مباشـرة وإعطاءهـا التسميات المناسبة:

يتم رسم الوحدة الوظيفية الرئيسة في قمة المخطط، ثم يتم التوسع تحتـها بالوحدات التابعة لها مباشرة، حيث يجب تجميع العمليات الموجودة في مخططـات التدفق إلى مجموعة عمليات الإدخال، ومجموعة عمليات الإخراج، ومجموعة المعالجة أو التحويل. وبناء على هذه المجموعات يتم رسم الوحدة الوظيفية التي سـيتم مـن خلالها استدعاء الوحدات الوظيفية الموجودة في مجموعة عمليات المعالجة ووضعـها مباشرة تحت الوحدة الرئيسة ثم نربط هذه الوحدة بالوحدة الرئيسية.

وبنفس الطريقة يتم رسم الوحدة الوظيفية المتعلقة بعمليات الإدخـــال إلى اليسار ليتم من خلالها استدعاء الوحدات الوظيفية الخاصة بالإدخـــال (القـــراءة والتدقيق وإظهار رسائل الأخطاء وغيرها)، وربطها أيضا بالوحدة الرئيسة.

وأخيراً يتم رسم الوحدة الوظيفية الخاصة بعمليـــات الإخـــراج إلى يمـــين الوحدة الوظيفية للمعالجة ثم ربطها مع الوحدة الرئيسية . وهكذا نكون قد رسمنـــا المستوى الأول والثاني في المخطط الهيكلي .

2) **إنشاء الوحدات الوظيفية للمستويات الدنيا** : وذلك برسم الوحدات الوظيفية الخاصة بعمليات المعالجة تحت الوحدة الوظيفية للمعالجة والموجودة في المستوى الثاني للمخطط . يتم تحديد هذه العمليات من المستوى الأدنى لمخططات تدفق البيانـــات الخاص بكل عملية من العمليات الرئيسية . ونكـــرر الطريقـــة نفســـها بالنســـبة للعمليـــات الموجودة ضمن مجموعة عمليات الإدخال ومجموعة عمليات الإخـراج . ونربط هذه العمليات بشكل متسلسل تحت بعضها أو بجانب بعضها بحسب عـــدد مستويات مخططات تدفق البيانات . فإذا كان لدينا مثلاً بالإضافة إلى مخطط تدفـــق البيانات العام ثلاث مستويات تفصيلية فإننا سنحتاج على الأغلب لثلاث مستويات أدنى للوحدات الوظيفية بالإضافة إلى المستويين الأول والثاني اللذين يمثلون الوحـــدة الوظيفية الرئيسية والوحدات التابعة لها مباشرة (غالباً يقابل هذا المستوى المخطـــط العام لتدفق بيانات النظام Top-Level DFD) .

3) **مراجعة المخطط الأولي الناتج وتدقيقه:**

وذلك بهدف تحسينه والتأكد من خلوه من الأخطاء واكتماله وجعله أكـــثر ملائمة لمتطلبات البنية الهرمية والتصميم الهيكلي، وغالباً خلال هذه المراجعة يتـــم

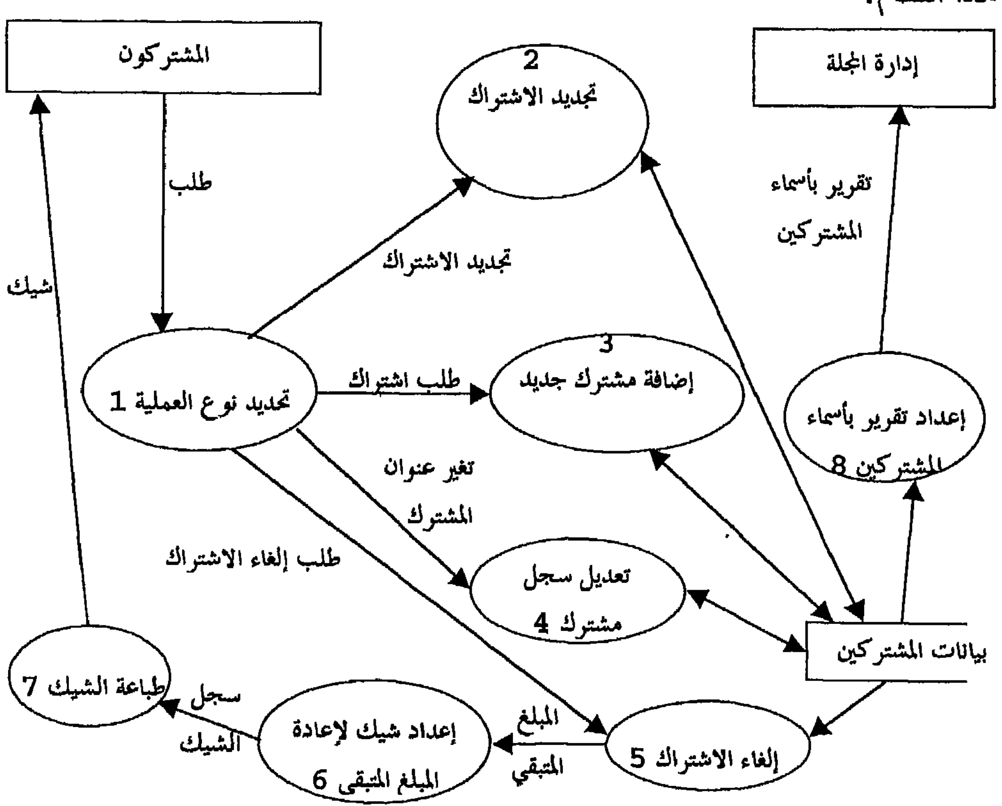
إدخال العديد من التعديلات على المخطط لتحسين جودته على النحو الذي سندرسه في الفقرة التالية والمتعلقة بجودة تصميم مخططات التدفق.

وفي نهاية هذه المراجعة يتم التوصل إلى المخطط الهيكلي النهائي.

4) مناقشة المخطط النهائي مع الإدارة والمستخدمين وبقية أعضاء الفريق المشارك في تطوير النظام، للتأكد من اكتماله واحتوائه على جميع الوظائف المطلوبة من النظام، واللازمة لتلبية احتياجات المستخدمين.

مثال لإعداد مخطط هيكلي:

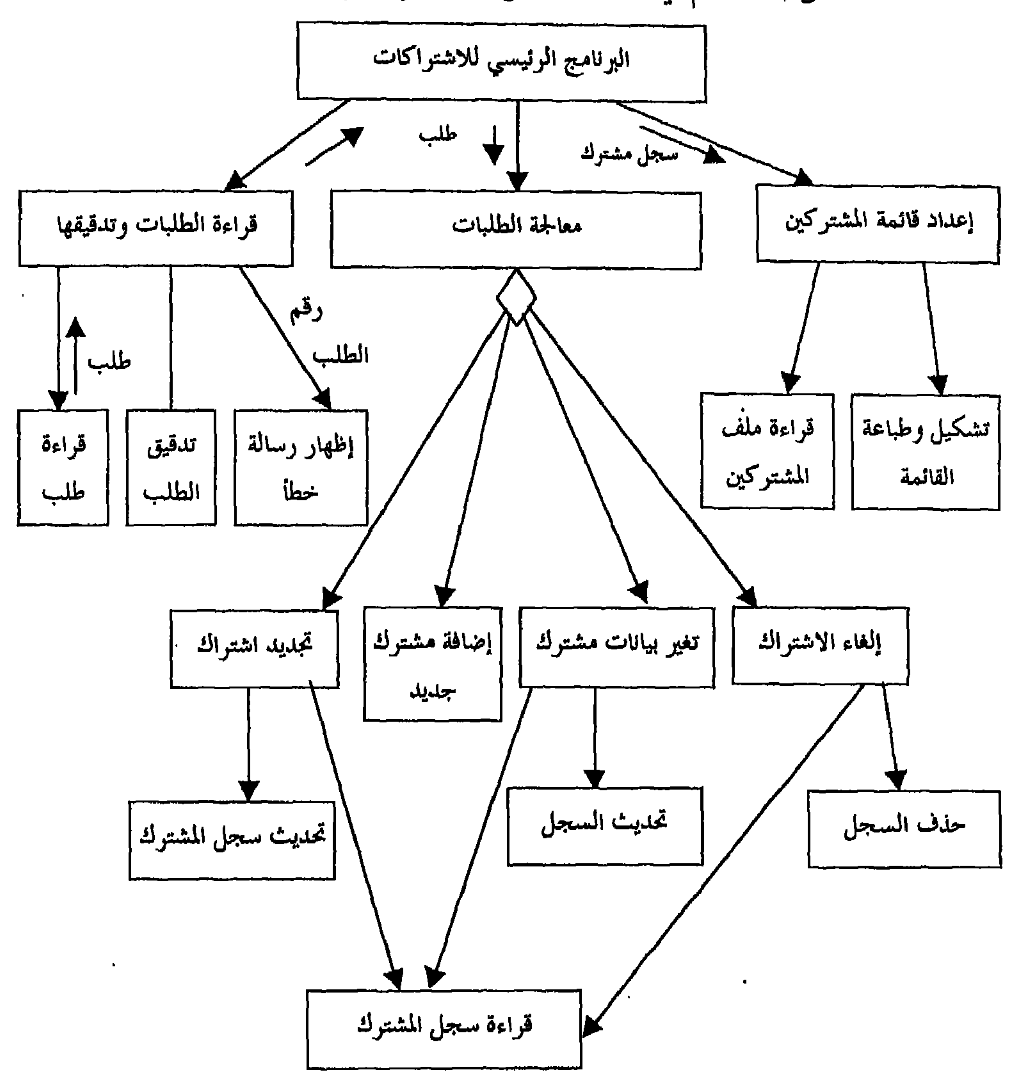
يبين الشكل (7.13) مخطط تدفق البيانات لنظام الاشتراكات في إحدى المجلات، وسنستخدم الخطوات المشار إليها أعلاه لإعداد المخطط الهيكلي لبرامج هذا النظام.



شكل (7 . 13) مخطط تدفق البيانات في نظام الاشتراكات

لتحويل المخطط أعلاه إلى مخطط هيكلي يمكن اتباع الخطوات التالية:

1- نرسم الوحدة الوظيفية الرئيسة، والوحدات التابعة لها مباشرة. تمثل الوحدة الرئيسة البرنامج الرئيسي للنظام، أما الوحدات التابعة لها مباشرة فتتعلق بالتحكم في المدخلات والمعالجة والمخرجات كما هو مبين أدناه.



شكل (8.13) المخطط الهيكلي لنظام الاشتراك (مثال)

2- إنشاء الوحدات الوظيفية للمستوى الأدنى: ويتم ذلك بإضافة ا لوحدات الوظيفية التي تتضمن العمليات اللازمة لقراءة الطلبات وتدقيقها وإظـــهار رسائل الأخطاء في حال وجودها، ثم العمليات المتعلقة بالمعالجـــة، وهـــذا يتضمن استدعاء الوحدة الوظيفية التي ستقوم بالعملية المطلوبة في ضوء نوع العملية، وأخيراً يتم إضافة الوحدات الوظيفية اللازمـــة لإعـــداد التقريـــر المطلوب وفي النهاية يمكن إضافة الأزواج وكتابة تسمياتها علـــى النحـــو المبـــين في الشكل (8.13).

3- مراجعة المخطط لتدقيقه والتأكد من خلوه من الأخطاء.

6. جودة المخططات الهيكلية

يتم تطوير المخططات الهيكلية باســـتخدام منهجيـــة التصميـــم الهيكلـــي Structured Design، التي تهدف بشكل عام إلى إعداد مخططات ذات خصـــائص تنسجم مع قواعد البرمجة الهيكلية، ولقياس مدى تطابق هذه المخططات مع معايـــير التصميم الهيكلي يستخدم مفهومان أساسيان هما تماســـك الوحـــدات الوظيفيـــة Module Cohesion وترابطها Module Coupling. وسنتعرف فيما يلي إلى هذيـــن المفهومين الهامين وكيفية استخدامهما لتحديد جودة المخططات الهيكلية.

1.4.3 تماسك الوحدة الوظيفية Module Cohesion

ويسمى هذا المقياس أيضاً قوة الوحدة الوظيفية Module Strength حيـــث يفترض هذا المفهوم أن تتكون كل وحدة وظيفية من مهمة واحدة فقط غير قابلـــة للتقسيم، والمخطط الهيكلي الجيد هو ذلك المخطط الذي يتكون من وحدات وظيفية متماسكة جيداً. تعتبر الوحدة الوظيفية متماسكة جيداً عندما تتضمن تنفيذ مهمـة أو

عملية واحدة فقط، أما إذا تضمنت الوحدة عدة مهام أو عمليات فـــان تماســكها يكون ضعيفاً. ولذلك لكي يضمن المصمم الوصول إلى مخطط جيد التصميم يجـــب أن يقوم بتقسيم النظام إلى عمليات واضحة ومحددة (خلال إعداد مخططات تدفـــق البيانات)، ثم تخصيص كل عملية منها بشكل وحدة وظيفية في المخطـــط الهيكلـــي للنظام.

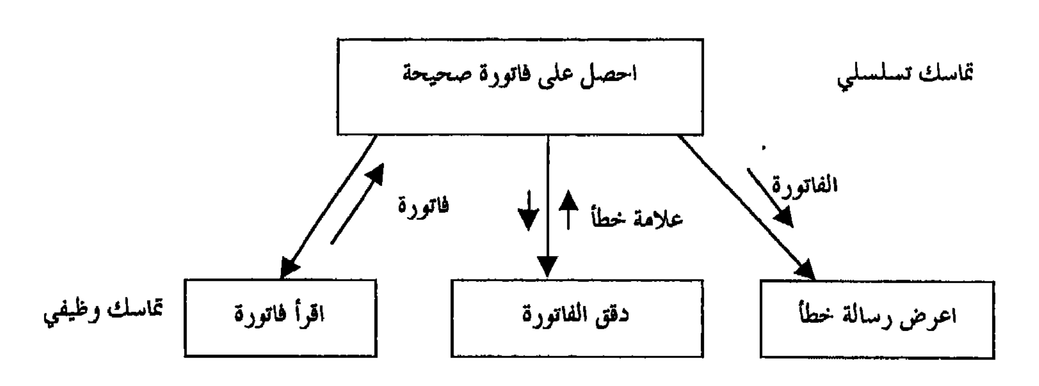
تتفاوت قوة أو تماسك الوحـــدات الوظيفيـــة في المخططـــات الهيكليـــة باختـــلاف **أنواع التماسك** الممكنة والتي سنستعرضها فيما يلي:

التماسك الوظيفي Functional Cohesion

تكون الوحدة الوظيفية متماسكة وظيفياً عندما تتضمن القيـــام بوظيفـــة (عملية) واحدة فقط، ويعتبر هذا النوع من أفضل أنواع التماسك ويجب الحـــرص على أن تكون جميع أو معظم الوحدات الوظيفية في المخطط الهيكلـــي متماســكة وظيفياً. ويحدث هذا التماسك غالباً في الوحدات الوظيفية الموجودة في المســتويات الدنيا للمخططات الهيكلية. أما الوحدات الوظيفية الموجودة في المستويات الأعلـــى فيكون لها غالباً عدة وحدات تابعة، وبالتالي فإنها تمثل غالباً أكثر من مهمة. ونظـــراً لكون الوحدات الموجودة في المستويات العليا لا تقوم بأي مهام تنفيذية، بل ينحصر عملها في التحكم بالوحدات التابعة واستدعائها لتنفيذ المهام المختلفة، فان تماسكها الوظيفي يكون أقل من تماسك الوحدات التابعة لها. وبشكل عام يجدر الانتبـــاه إلى انه عندما تكون الوحدات الوظيفية الموجودة في المستوى الأدنى للمخططات الهيكلية غير متماسكة وظيفياً، فان هذا يعني إمكانية تقسيمها إلى العمليات المكونـــة لهـــا وإضافة مستويات أدنى في هذا المخطط.

التماسك التسلسلي Sequential Cohesion

يحدث هذا النوع من التماسك عندما تتضمن الوحدة الوظيفية عدة مهام متسلسلة منطقياً، أي تكون مخرجات المهمة الأولى هي مدخلات للمهمة التالية، وهكذا فمثلاً في الشكل (9.13) تكون الوحدة الوظيفية احصل على فاتورة صحيحة متماسكة تسلسلياً، نظراً لأنها تقوم باستدعاء الوحدة الوظيفية" قراءة فاتورة" التي ينتج عنها وصول بيانات حركة ما إلى هذه الوحدة المستدعية، ثم ترسلها مباشرة كمدخلات للوحدة الثانية والمتعلقة بتدقيق الفاتورة. وهكذا فان الوحدة المسماة " احصل على فاتورة صحيحة" تقوم بمهمتي استدعاء متسلسلين ولذلك يسمى تماسكها بالتماسك التسلسلي. أما الوحدات التابعة لها فانه من الواضح أنها متماسكة وظيفياً لأن كل منهما يقوم بمهمة واحدة فقط.



شكل (9.13) التماسك الوظيفي والتسلسلي

التماسك الإتصالي: Communicational Cohesion

يحدث التماسك الإتصالي عندما تتضمن الوحدة الوظيفية عدة مهام أو وظائف تتبادل بيانات بين بعضها البعض، فمثلاً جميع العمليات المتعلقة بالتعامل مع ملف ما يمكن جمعها في وحدة وظيفية واحدة تقوم بقراءة الملف ومعالجته وتحديثه، وفي هذه الحالة تكون هذه الوحدة الوظيفية متماسكة إتصالاتياً، اي تتضمن عدة مهام تشترك معاً باستخدام نفس البيانات.

ويختلف هذا النوع من التماسك عن التماسك التسلسلي، بأن تسلسل تنفيذ العمليات في الوحدة الوظيفية المتماسكة تسلسلياً يكون مهماً وضرورياً، أما في الوحدات الوظيفية المتماسكة اتصالاتياً فانه يمكن تنفيذ المهام المختلفة الموجودة ضمنها وفق أي ترتيب كان، ومن الواضح أن هذا النوع من التماسك كما هو الأمر في التماسك التسلسلي ممكن في الوحدات المشرفة أي تلك التي يوجد لديها وحدات تابعة لها.

التماسك الإجرائي Procedural Cohesion

يحدث هذا النوع من التماسك عندما تقوم الوحدة الوظيفية بتمرير إشارات التحكم بدلاً من البيانات من وحدة وظيفية إلى أخرى، وهذا يعني أن الوحدة الوظيفية تتضمن عدة مهام متسلسلة ولكن لا تشترك معاً بأية بيانات (التماسك الإتصالاتي) ولا تمرر أية بيانات فيما بينها (التماسك التسلسلي). وهكذا فان الوحدة الوظيفية المتماسكة إجرائيا تتضمن استدعاء متسلسل لعدة وحدات وظيفية تابعة لها دون أن تكون هناك أية علاقة (بيانات مشتركة) بين هذه الوحدات. وهذا

النوع من التماسك يعتبر ضعيفاً ويجب عدم استخدامه إلا عندما يتعلق الأمر بالوحدة الوظيفية الرئيسة في المخطط التي يمكن أن تكون متماسكة إجرائيا.

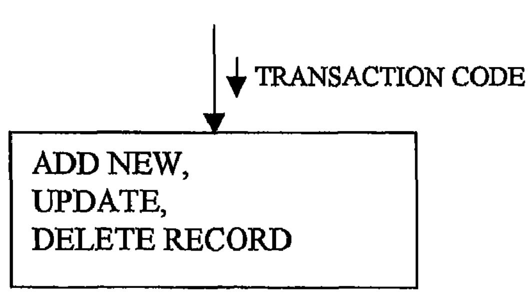
التماسك المؤقت Temporal Cohesion

تكون الوحدة الوظيفية متماسكة مؤقتاً إذا كانت تتضمن عدة عمليات لا يوجد بينها أي علاقة (لا بيانات مشتركة ولا تسلسل في التنفيذ) سوى كونها يمكن أن تنفذ في وقت واحد (في نفس الوقت).

ومن أمثلة هذا النوع من التماسك تلك الوحدات الوظيفية التي تتضمن عمليات فتح الملفات وإظهار الشاشات وكتابة العناوين وتصفير العدادات وغير ذلك من الإجراءات التي لا علاقة بينها سوى كونها تتم في نفس الوقت، وهذا النوع من التماسك غير مقبول ويجب تجنب استخدامه، وتوزيع مثل هذه العمليات لتتم ضمن الوحدات الوظيفية الأخرى عند الحاجة إليها.

التماسك المنطقي Logical Cohesion

تكون الوحدة الوظيفية متماسكة منطقياً عندما تتضمن عدة وظائف أو عمليات، ويتم عند كل استدعاء لها تنفيذ واحدة فقط من هذه العمليات في ضوء إشارة أو راية Flag يتم إرسالها من الوحدة المشرفة مع كل عملية استدعاء، كما هو مبين أدناه.



ويعتبر هذا النوع من أسوء أنواع التماسك ويجب عدم استخدامه على الإطلاق، فبدلاً من الوحدة الوظيفية أعلاه يجب استخدام ثلاث وحدات وظيفية، تقوم كل منها بإحدى الوظائف الثلاث أعلاه (إضافة – تعديل – حذف).

كما قد يتضمن التماسك المنطقي تجميع عدة عمليات ذات طبيعة متماثلة في وحدة وظيفية واحدة، فمثلاً جميع عمليات قراءة الملفات في وحدة وظيفية واحدة، أو جميع عمليات تحرير السجلات المستخدمة في النظام في وحدة وظيفية واحدة. ومن الواضح انه عند استدعاء مثل هذه الوحدات يجب أن يتم تمرير رمز أو علامة بنفس الأسلوب السابق لتحديد العمليات المطلوبة عند كل استدعاء، وهذا النوع أيضاً كسابقه غير مقبول في المخططات الهيكيلية.

التماسك المتزامن أو المتوافق Coincidental Cohesion

وهو أيضاَ من أسوء أنواع التماسك وغير مقبول لكون الوحدة الوظيفية المتماسكة بهذا الأسلوب تتضمن عدداً من العمليات غير المرتبطة مع بعضها البعض.

ونذكر أخيراً بأنواع التماسك المقبولة في المخططات الهيكلية وهي:

- التماسك الوظيفي
- التماسك التسلسلي
- التماسك الإتصالاتي
- التماسك الإجرائي (الوحدة الرئيسة)

أما الأنواع الأخرى فغير مقبولة ويدل وجودها في المخطط على وجود ضعف في التصميم.

2.6 ترابط الوحدات الوظيفية Module Coupling

بالإضافة إلى مقياس التماسك الذي يعبر عن القوة الداخلية للوحدات الوظيفية، يستخدم مقياس الترابط الذي يقيس شدة العلاقة بين هذه الوحدات، وهذان المقياسان متكاملان من حيث الجوهر، فالوحدة الوظيفية المتماسكة بقوة تكون أكثر استقلالية من تلك الوحدات الأقل تماسكاً، ولذلك فان العلاقة التي تربط الوحدات الوظيفية المتماسكة جيداً يجب أن تكون ضعيفة Loose Coupling وهذه هي الحالة المرغوبة بالنسبة للترابط، فالترابط الجيد هو الترابط الضعيف Loose.

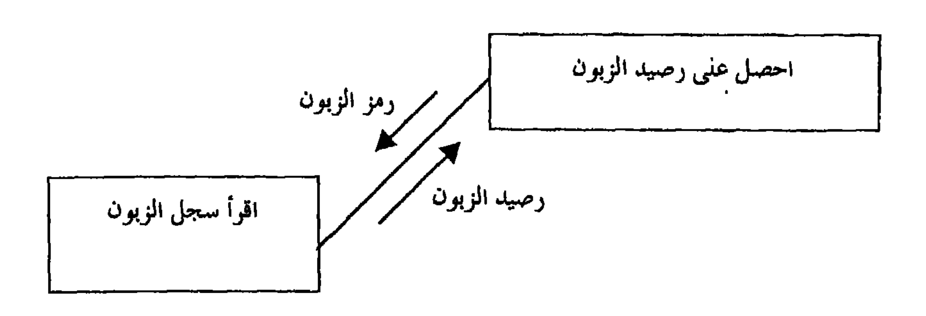
وهذا ينسجم أيضا مع مبدأ إعادة الاستخدام Reuse الذي يعتبر من مزايا التصميم الهيكلي حيث يمكن استخدام الوحدة الوظيفية الواحدة في أكثر من برنامج وفي أكثر من نظام، وكلما كانت الوحدة الوظيفية اكثر استقلالاً (متماسكة وظيفياً) كما أمكن إعادة استخدماها بشكل أسهل وعلى نطاق أوسع.

وسنتعرف فيما يلي إلى الأنواع المختلفة للترابطات الممكنة بين الوحدات الوظيفية، وذلك أيضا بشكل متسلسل حيث سنبدأ بالترابط الأفضل ثم الأقل جودة فالأسوأ.

الترابط البياناتي Data Coupling

تكون الوحدتين الوظيفيتين مترابطتان بياناتياً عندما تقومان بتمرير البيانات فقط فيما بينها.

ويعتبر الترابط البياناتي افضل أنواع الترابط، ويبين الشكل (8.13) مثالاً للترابط البياناتي بين الوحدات الوظيفية.



شكل (10.13) الرابطة البياناتية بين وحدتين وظيفيتين

وهكذا نجد في الشكل أعلاه أن الوحدتين أعلاه تتبادلان حقول البيانات فيما بينهما، فالوحدة المشرفة تستدعي الوحدة التابعة وتزودها بعنصر بيانات هو "رمز الزبون"، فتقوم الوحدة التابعة بقراءة سجل ذلك الزبون وتعيد عنصر بيانات آخر هو " رصيد الزبون" إلى الوحدة المستدعية.

الترابط الكتلي: Stamp Coupling

تكون الوحدتان الوظيفيتان مترابطتان كتليا عندما تقومان بتمرير بنية بيانات (مؤلفة من مجموعة عناصر) بين بعضهما البعض فمثلا عندما تقوم الوحدة الوظيفية المستدعية بتمرير " سجل الفاتورة" أو " سجل الدفعة" إلى الوحدة التابعة، والتي بدورها تقوم بتدقيق هذا السجل ثم تعيده إلى الوحدة المشرفة، فان مثل هذا الترابط يكون ترابطا كتليا. وهو مقبول ولكن يجب الحذر من استخدامه بكثرة، حيث يمكن أن يكون هناك العديد من حقول البيانات التي لا تقوم الوحدة الوظيفية المستدعاه باستخدامها، ولذلك لا ضرورة لإرسال كامل السجل، بل يجب الاقتصار على إرسال الحقول الضرورية فقط.

الترابط الرقابي Control Coupling

تكون الوحدتان الوظيفيتان مترابطتان رقابياً عندما تقوم إحداهما بالتحكم في المنطق الداخلي للوحدة الوظيفية الأخرى، ويتم هذا التحكم عادة من خلال تمرير راية أو إشارة Flag، فالراية إذن هي رابطة تحكم أو رقابة، ويمكن تقييم هذا النوع من الروابط من خلال ما يلي:

أ) **اتجاه الرابطة:** عندما تتجه رابطة التحكم (الراية) من الأسفل إلى الأعلى، أي من الوحدات التابعة إلى الوحدة المشرفة، فإنها تكون مقبولة لكونها تزود الوحدة الوظيفية المشرفة بمعلومات هامة مثل وجود خطأ أو الوصول إلى نهاية الملف أو غير ذلك،فتقوم الوحدة المشرفة بناء على هذه المعلومات بتحديد العمليات (الوحدات الوظيفية) المطلوب تنفيذها في ضوء هذه الحالات، ولذلك فان استخدام الرايات المتجه من الأسفل إلى الأعلى يعتبر أمرا طبيعيا ومقبولاً.

أما الرايات المتجه من الأعلى إلى الأسفل، كتلك التي تشير إلى نوع الحركة المطلوبة (مثلا: سحب، إيداع، تحويل من حساب إلى آخر) فإنها تعتبر بمثابة أمر مباشر من الوحدة الوظيفية المشرفة إلى الوحدة الوظيفية التابعة، وهذا يعني أن الوحدة المشرفة تعرف تفاصيل العمليات الموجودة في الوحدة التابعة، وهذا يناقض مبدأ استقلالية عمل الوحدات ولذلك يجب الإقلال من استخدام هذه الرايات المتجهة نحو الأسفل قدر الإمكان.

ب) **هدف الرابطة:** عندما تستخدم الراية للدلالة على حالة معينة فإنها تسمى راية وصفية Descriptive flag ، ومثل هذا النوع من الرايات مقبول إذا كانت متجهة من الأسفل إلى الأعلى كما أشرنا أعلاه.

أما إذا كان الهدف من استخدام الراية تبيان ما هو مطلوب عمله فإنها تسمى راية تحكم أو رقابة Control Flag ومثل هذه الرايات غير مقبولة سواء كانت متجهة الأعلى للأسفل أو العكس.

وتجدر الإشارة أيضا إلى انه في بعض الحالات قـــد يضطـــر المصمـــم استخدام الرايات الوصفية المتجهة إلى الأسفل أو رايات التحكم، وهذا يمكن قبو ضمن أضيق الحدود.

الترابط المشترك Common Coupling

ويسمى أيضا ترابط البيئة المشتركة، حيث يمكـــن أن ترتبـــط وحدتـ وظيفيتان من خلال رابطة مشتركة بينهما، وقد تكون مثلاً استخدام منطقة عمل الذاكرة أو ملف يتم استخدامه بشكل مشترك من قبل هاتين الوحدتين الوظيفيتـ وهذا النوع من الترابط يجب تجنبه قدر الإمكان بالرغم من كونه قد يكون الطر المناسبة للتعامل مع حالات معينة.

ترابط المحتوى Content Coupling

تكون وحدتان وظيفيتان مترابطتان من حيث المحتوى إذا كانت إحداهـ تقوم بالرجوع إلى محتوى الوحدة الوظيفية الأخرى، ومثل هذا الـــترابط يســ للوحدة المشرفة بتغيير أو تعديل برنامج الوحدة المستدعاة (الوحـــدة التابعـــة استخدام بيانات معرفة داخلياً ضمن هذه الوحدة، كما تسمح لإحدى الوحدتـ الوظيفيتين بالتفرع إلى الوحدة الوظيفية الأخرى. وهذا النوع من الترابط مرفـ تماماً ويجب تجنبه وعدم السماح به بأي حال من الأحوال، لكونه دليل واضح ع ضعف التركيبة البنيوية أو الهيكلية للنظام.

3.6 معايير تقييم شكل المخطط الهيكلي:

بالإضافة إلى مقاييس التماسك والترابط المذكورة أعلاه، يجب أيضاً تقييم المخطط من حيث شكله العامMorphology وذلك باستخدام المؤشرات التالية:

أ) مدى الرقابة Span of Control

ويقصد به عدد الوحدات الوظيفية التابعة مباشرة لوحدة وظيفية معينة، ويفضل أن يتراوح هذا العدد بين خمس وسبع وحدات للمشرف الواحد، ولكن في المخططات المتعلقة بمعالجة الحركات Transaction Processing يمكن أن يكون عدد هذه الوحدات مساو لعدد أنواع هذه الحركات.

ب) عدد الوحدات التي يمكن أن تستدعي وحدة معينة

ويسمى هذا المعيار أيضا Fan-in ، وهو عكس المعيار السابق، ويدل على مستوى إعادة استخدام الوحدات الوظيفية التابعة. لذلك فكلما كان هذا العدد كبيراً كلما دل ذلك على أن الوحدة الوظيفية تستخدم من قبل عدة وحدات مشرفة، مما يجعل المخطط الهيكلي أكثر كفاءة.

جـ) مجال الرقابة Scope of Control

ويقصد به عدد الوحدات الوظيفية التابعة لوحدة وظيفية معينة (سواء كانت هذه التبعية مباشرة أو غير مباشرة). وكلما كان هذا العدد كبيراً كلما دل ذلك على وجود مخطط هيكلي مصمم جيداً.

د) مجال التأثير Scope of Effect

ويقصد به عدد الوحدات الوظيفية التي يمكن أن تتأثر نتيجة لقرار معين موجود في وحدة وظيفية ما ، والتصميم الجيد للمخططات الهيكلية يتطلب أن يكون

هذا المجال محدوداً جداً حيث لا يجب أن تتأثر عمليات الوحدات الوظيفية بنتائج أو مخرجات قرارات موجودة في وحدات أخرى.

4.6 التحليل Factoring

تستخدم هذه العملية أثناء إعداد المخططات الهيكلية للنظام، ويقصد بالتحليل تقسيم الوحدة الوظيفية إلى وحدتين أو اكثر تقعان في المستوى الأدنى، وربطها مع وحدة مشرفة في المستوى الأعلى ، تساعد هذه العملية في:

- تصغير حجم الوحدات الوظيفية وتبسيط تعقيداتها.
- تسهيل وتسريع فهم عمليات النظام.
- توفير إمكانية افضل للوصول إلى وحدات وظيفية يمكن إعادة إستخدامها لأكثر من مرة، فالوظائف المتماثلة يتم تصميمها ضمن وحدات وظيفية مستقلة يمكن استدعائها من قبل جميع الوحدات الوظيفية التي تحتاج إليها.
- الفصل الواضح بين الوحدات الوظيفية الإدارية Management Modules والتي توجد عادة في المستويات العليا للمخطط الهيكلي ، والوحدات الوظيفية التنفيذية الموجودة في المستوى الأدنى لهذه المخططات .

وهكذا فإنه عند تصميم المخطط الهيكلي لبرامج النظام يتم تخصيص الوحدات الوظيفية في المستويات الدنيا للمخطط للعمليات التنفيذية (المستوى التفصيلي الأدنى لمخططات تدفق البيانات) ، وتكون هذه الوحدات متماسكة وظيفياً . أما الوحدات الوظيفية في المستويات العليا للمخطط فتضمن المهام الإدارية ويكون تماسكها تسلسلياً أو كتلياً .

ويجب أثناء التحليل تجنب أو معالجة المشاكل التي يمكن أن تنشأ عما يسمى بذاكرة الحالة State Memory . فعندما يتم استدعاء الوحدة الوظيفية التابعة تنتقل إلى ذاكرة الحاسوب، وبعد الانتهاء من تنفيذها تزول من الذاكرة ويعود التحكم إلى الوحدة المشرفة . وفي حال استدعاء هذه الوحدة مرة ثانية تنتقل نسخة جديدة منها إلى الذاكرة ، وكأنها تستدعي للمرة الأولى . وفي بعض الحالات قد تحتاج الوحدة الوظيفية المستدعاة إلى الاحتفاظ ببعض فقرات بيانات من الاستدعاءات السابقة . فمثلاً عندما يتم استدعاء وحدة وظيفية لطباعة تذييل التقرير تحتاج هذه الوحدة إلى معرفة (تذكر) رقم آخر صفحة تمت طباعتها . أو عند استدعاء الوحدة الوظيفية الخاصة بطباعة الفاتورة فإنها تحتاج لمعرفة رقم آخر فاتورة تم طباعتها لتحديد رقم الفاتورة التالية ، وهكذا . ولكي لا يضطر المبرمج إلى معالجة مثل هذه التعقيدات في البرنامج ، والتي تتطلب الاحتفاظ بهذه البيانات بين الاستدعاءات المتتالية للوحدات الوظيفية ، يجب تجنب مثل هذه الحالات ومعالجتها بجعل الوحدات المشرفة مسؤولة عن متابعة هذه القيم وتمريرها إلى الوحدات التابعة عند كل استدعاء. فالوحدات المشرفة موجودة بشكل دائم في الذاكرة حتى انتهاءها من إنجاز وظيفتها. كما يمكن اللجوء إلى عملية التحزيم ووحدات التحميل Load module لمعالجة هذه المشكلة.

7. لغات الجيل الرابع Fourth Generation Languages

هي لغات برمجة غير إجرائية Non - Procedure Oriented مما يميزها عن اللغات الاجرائية التي تنتمي إلى الجيل الثالث مثل لغة BASIC و COBOL و PASCAL و غيرها.

تمتاز لغات الجيل الرابع بسهولة استخدامها User Friendly ويمكن أن تتضمن المكونات الآتية:

- لغة استعلام بنيوية Structured Query Language
- مولدات التقارير Reports Generators.
- مولدات الشاشات والنماذج Forms and Screen Generators
- مولدات التطبيقات Application Generators.

تساعد هذه اللغات في زيادة سرعة وإنتاجية عمليات تصميم وتنفيذ أنظمة المعلومات. وعند استخدام المحلل أو المصمم لهذه اللغات، يقوم بتحديد احتياجاته أي تحديد أشكال التقارير والشاشات والاستعلامات والتطبيقات المطلوبة، ويقوم الحاسوب بتحديد الإجراءات اللازمة لتنفيذ هذه الاحتياجات، وهكذا يمكن بفضل هذه اللغات كتابة بضعة أوامر أو عبارات وإدخالها إلى الحاسوب حيث تقوم المترجمات بتحويلها إلى آلاف التعليمات المكتوبة بلغة الآلة.

أما أهم لغات الجيل الرابع المنتشرة حالياً فهي Focus و RAMIS و ORACLE و IDEAL وغيرها.

وهكذا يمكن باستخدام هذه اللغات تحسين إنتاجية المبرمجين وتقليص الوقت اللازم للبرمجة بشكل كبير، كما أن سهولة استخدماها تمكن المستخدمين العاديين من بناء تطبيقاتهم بأنفسهم دون الرجوع إلى أخصائيي الأنظمة، ويمكن استخدام هذه اللغات في الحواسيب المايكروية، ويتوقع أن يلاقي استخدامها انتشاراً واسعاً نظراً للتطورات السريعة التي تشهدها تقنية الحواسيب والاتجاه نحو تقييس هذه الأدوات وزيادة كفاءة عملها.

وأخيرا تجدر الإشارة إلى أن هذه اللغات تقدم لمحلل الأنظمة مساعدة كبيرة من خلال تمكينه من إنتاج النماذج التجريبية Prototypes المختلفة ودراستها وتجربتها بالتعاون مع المستخدم.

أسئلة الفصل

1- عدد واشرح الخصائص التي تميز البرامج الحاسوبية الجيدة؟

2- ما هي المواصفات الوظيفية وكيف يتم تحديدها؟

3- ما المقصود بالبرمجة الهيكلية وما هي البنى الأساسية المستخدمة فيها؟

4- اشرح خطوات تصميم برامج النظام؟

5- ما هي عناصر التصميم التفصيلي للوحدات الوظيفية؟

6- ما هي المخططات الهيكلية واشرح عناصرها؟

7- اشرح التراكيب الممكنة للمخططات الهيكلية؟

8- اشرح خطوات إعداد المخططات الهيكلية؟

9- ما هي أهم المعايير المستخدمة لقياس جودة المخططات الهيكلية؟

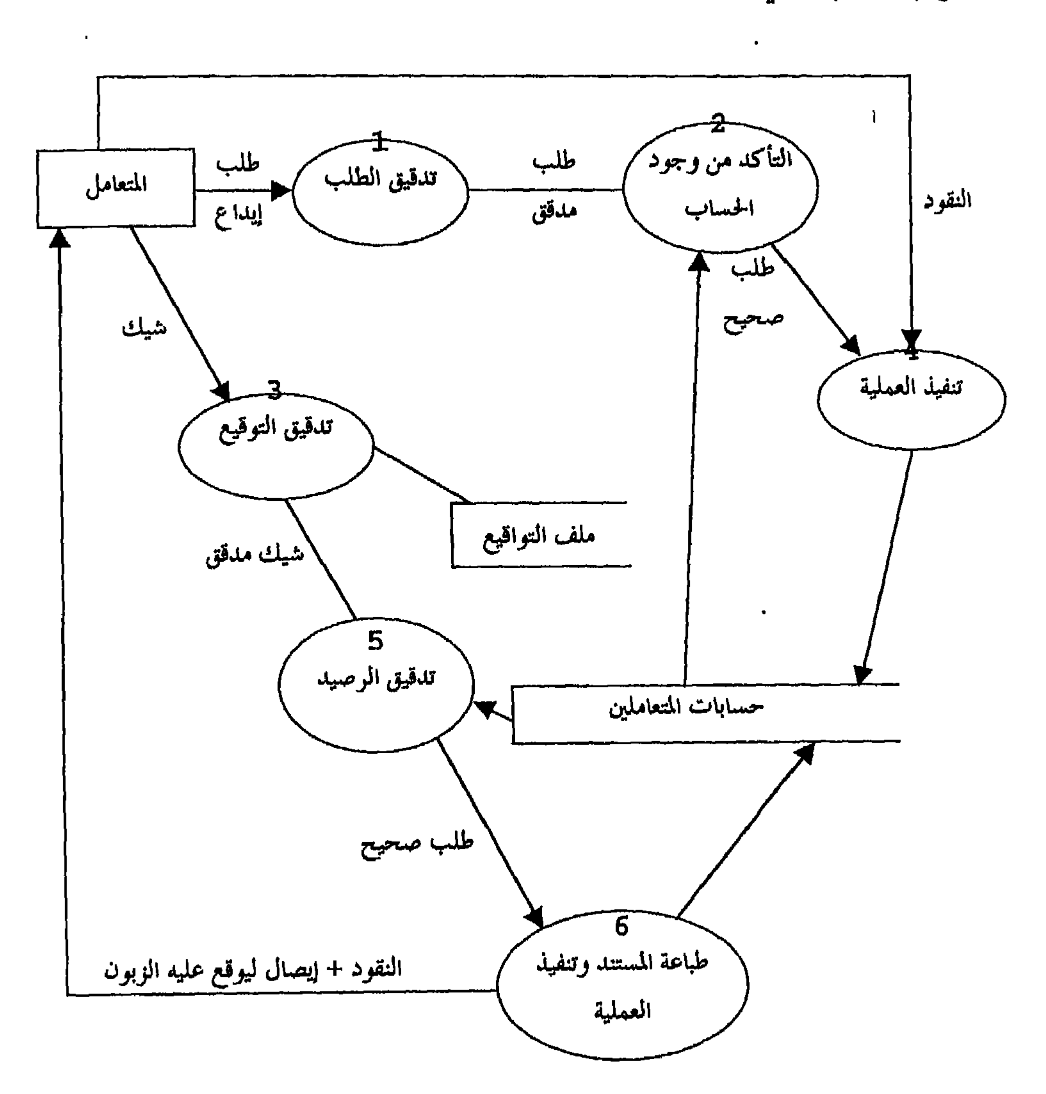
10- ما المقصود بتماسك الوحدات الوظيفية ؟ واشرح أنواع التماسك الممكنة وحدد الأنواع المقبولة وغير المقبولة؟

11- ما المقصود بترابط الوحدات الوظيفية؟ واشرح أنواع الترابط الممكنة بين الوحدات وحدد المقبول منها؟

12- ما هي ميزات لغات الجيل الرابع بالمقارنة مع لغات البرمجة التقليدية؟

تمارين

1- ارسم المخطط الهيكلي لنظام عمليات الحسابات الجارية في البنك المبين في الشكل (11.13) التالي:



2- ارسم المخطط الهيكلي لنظام معالجة دفعات الزبائن الـــــذي يتضمـــن العمليات التالية:

1- قراءة العمليات المتعلقة بدفعات الزبائن وتدقيقها وإظهار رسالة خطـــــأ في حال وجود أي أخطاء فيها.

2- قراءة سجلات الزبائن لكل عملية (دفعة) متسلسلة رســـــالة خطـــأ في حـــال عدم وجود السجل المطلوب.

3- حساب الرصيد الجديد في ضوء مبلغ الدفعة.

4- تحديث سجل الزبون.

5- طباعة ميزان مراجعة بالدفعات التي تمت معالجتها.



الإتجاهات الحديثة في تحليل وتصميم الأنظمة



الفصل الرابع عشر

التحليل والتصميم الموجه للكيانات

Object – Oriented Analysis and Design

يزاد الاهتمام بهذا المدخل الجديد لتحليل وتصميم أنظمة المعلومات وينتشر استخدامه بشكل واسع مع تطور لغات البرمجة الحديثة مثل C++ ولغات البرمجة المرئية Visual Programming وغيرها من اللغات المصممة لهذا الغرض. ويدرس هذا الفصل المفاهيم الأساسية لهذا المدخل الحديث، وكيفية الاستفادة منها في تحليل وتصميم الأنظمة ، والتوجه نحو نمذجة الكيانات (الأشياء) Object Modeling بدلاً من نمذجة العمليات والبيانات بصورة منفصلة، كما هو الحال في طرق النمذجة المستخدمة حالياً.

1. مفاهيم التحليل والتصميم الموجه نحو الكيانات :

يقوم مدخل التطوير الموجه للكيانات Object-Oriented Development على المفاهيم التالية :

- جمع كل عملية من عمليات النظام مع البيانات الخاصة بها ، ومع تدفقات هذه البيانات من العملية واليها ، في مجموعة واحدة تسمى **كياناً أو شيئاً Object** .
- نمذجة هذه الكيانات ، أي تمثيلها بشكل كينونات مستقلة ، يمكنها التفاعل فيما بينها لتشكل النظام الذي تجري دراسته .

- تطوير النظام من خلال تطوير الكيانات Objects المكونة له . وهـذا يسرع عمليات التحليل والتصميم حيث لا حاجة لتحويل النموذج الـذي تم التوصل إليه خلال عمليات التحليل إلى نموذج تنفيذي .
- يدعم هذا المدخل استخدام المعلومـات ذات الأشـكال المتعـددة (الوسائط المتعددة) ، ولا يقتصر على استخدام البنى التقليدية للسجلات.

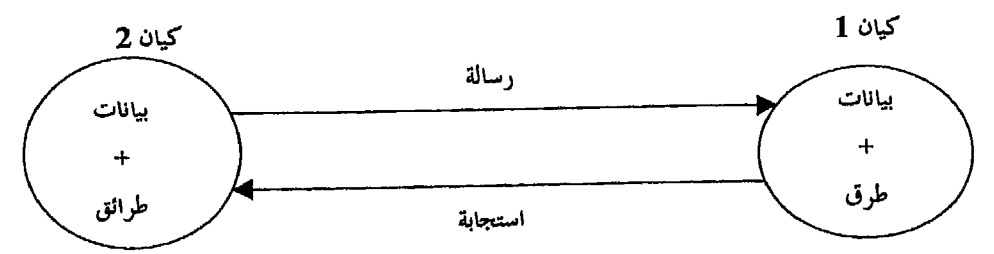
وبشكل عام ، فإنه بفضل هذا المدخل الجديد في تطوير الأنظمة لن تكـون هنالك حاجة في المستقبل لتطوير أنظمة معلومات كبيرة ، فبدلاً من ذلـك سـتتم عملية التطوير من خلال تحديد الكيانات التي يتكون منها النظام والتي تمثـل كمـا ذكرنا كينونات (أشياء) مستقلة لها أهدافها الخاصة بها ضمن إطار النظام الكلـي. وسيتم تطوير هذه الكيانات كل على حدة وبشكل متدرج وحسب الضرورة.

إن الفكرة الرئيسة التي يقوم عليها هذا المدخل الحديث في تطوير الأشـياء هي فكرة الكبسله **Encapsulation** أو التجميع في كل واحـد هـو **الكيـان أو الشيء Objects**.

وهكذا فإن الكيان يتم فيه تجميع البيانات مع العمليات . ويمكن تعريـف عملية الكبسلة هذه بأنها جمع البيانات والطرق في شيء واحـد ، حيـث يقصـد بالطرق هنا العمليات أي البرامج الموجودة ضمن الشيء.

وتتفاعل الكيانات أو تتصل مع بعضها البعـض مـن خـلال **الرسـائل Messages** التي تتبادلها فيما بينها. حيث يمكن للرسالة أن تُنشـط عمليـة معينـة (طريقة) لدى الشيء الذي يستلم هذه الرسالة، مما يؤدي إلى تنفيذ البرنامج الخـاص بهذه العملية ، والذي قد ينتج عنه استجابات معينة يمكن أن تكون بدورها رسـائل توجه للكيانات أو الأشياء الأخرى.

ويبين الشكل (1.14) مخططاً لتفاعل الكيانات مع بعضها البعض .



شكل (1.14) تفاعل الكينونات فيما بينها .

وبالإضافة إلى الكبسلة فإن هذا المدخل يقوم أيضاً على مبدأ هام آخر هو **استقلالية الكيانات** Object Autonomy، الذي يتطلب تصميم الكيان بصورة مستقلة عن الكيانات الأخرى في النظام. حيث لا يحتاج المحلل كما في أساليب التطوير السابقة ، إلى إعداد نماذج منفصلة لكل من العمليات والبيانات . بل يركز اهتمامه في نموذج واحد للكيان الذي يقوم بتصميمه والذي يضمنه جميع البيانات الخاصة بهذا الكيان والعمليات التي يمكن أن يقوم بها . إن مبدأ استقلالية الكيانات مهم جداً حيث يجعل إجراء التغييرات في النظام أكثر سهولة . كما يساعد في تطبيق مبدأ هام آخر هو **إعادة الاستخدام Reuse** ، حيث يمكن بسهولة تضمين هذه الكيانات أو الأشياء المستقلة في الأنظمة الجديدة كلما لزم الأمر. ولذلك يجب تصميم هذه الكيانات آخذين بالاعتبار ضرورة توفير المتطلبات اللازمة لإعادة استخدامها عند تصميم أنظمة أخرى في المستقبل .

وأخيراً تجدر الإشارة إلى وجود عدد من الأنظمة البرمجية (لغات برمجة وأنظمة إدارة قواعد البيانات) التي تدعم بشكل مباشر استخدام هذا المدخل لتطوير أنظمة المعلومات . وهذا يسهل عملية تحويل النماذج التي يتم التوصل إليها خلال

مرحلة التحليل (مثلاً E-RD و DFD) إلى كيانات على المستوى التنفيذي . مما يسمح بالتحويل المباشر لهذه النماذج إلى كيانات وتنفيذها .

وخلاصة القول أن التحليل والتصميم الموجه للكيانات OOD يمثل طريقة جديدة لنمذجة عناصر النظام ، تختلف جذرياً عن الطرق السابقة . فالنظام هنا هو عبارة عن مجموعة من الكيانات المستقلة ، حيث لكل منها وظيفته المحددة ، وترتبط معاً بطريقة ما لتحقيق الهدف العام للنظام .

2. بنية الكيان Object Structure :

تتألف بنية الكيان بشكل عام ، كما هو مبين في الشكل (2.14) من هياكل البيانات Data Structure ، والعمليات التي يتم من خلالها معالجة هذه البيانات .

Properties	Methods
DEPARTMENT-ID : ABC;	ADD-PERSON();
MANAGER: ref. PERSONS;	DEL-PERSON();
BUDGET : 15000 JD;	CREATE-TASK();
PEOPLE – ASSIGNED : ref. PERSONS(N);	CHANGE-BUDGET();
PROJECTS: ref. PROJECTS(N);	ADD-PROJECT();
TASKS: ref. TASK;	CLOSE-PROJECT().

شكل (2.14) بنية الكيان .

نلاحظ من الشكل (2.14) أعلاه أن بنية هذا الكيان البسيط تتألف من مجموعة **الخصائص** أي البيانات الخاصة بالكيان ، ومجموعة **الطرق** أي العمليات أو البرامج التي يمكن أن تحدث تغييراً في خصائص الكيان.

ونلاحظ أيضا أن خصائص الكيان يمكن أن تكون قيم بسيطة مثل رمز المديرية واسم المديرية والميزانية المخصصة لها ، بينما تتضمن الخصائص الأخرى مراجع References إلى كينونات أخرى مثل الأفراد والمهام والمشاريع. ونلاحظ أيضاً أن المرجع الخاص بالخاصية المتعلقة بالأفراد العاملين في الإدارة تشير إلى اسم الكيان (وهو الأفراد) والى حرف N بعده يشير إلى أن هذه الخاصية يمكن أن تتضمن عدداً من المراجع والسجلات وليس مرجعاً واحداً فقط ، كما هو الحال في خاصية " المدير " حيث لا يمكن أن يكون للمديرية سوى مديراً واحداً. أما الخاصية المتعلقة بالمهام فإنها تتضمن إرجاعاً لكيان آخر هو " مهام المديرية " ، ومن الواضح أن هذا الكيان يعتبر جزءاً من كيان المديرية ، أي لا يمكن أن تكون المهام موجودة، إذا لم يكن الكيان الأكبر (المديرية) موجوداً . وهكذا يمكن استخدام تعبير " يتضمن " Composite للإشارة إلى أن كيان ما (مثلاً المديريات) يمكن ان يتضمن كيانات أخرى (مثلاً مهام المديرية) . وهذا يعني أنه يمكن أن يتضمن الكيان كيانات أخرى .

أما المجموعة الثانية فهي العمليات (الطرق أو البرامج) التي يمكنها تغيير قيم خصائص الكيان . ونلاحظ في الشكل (2.14) وجود ستة طرق تستخدم لإضافة موظف إلى المديرية أو لحذف موظف منها أو لإدخال مهمة جديدة أو لتغيير الميزانية أو لإضافة وإغلاق المشاريع التي تشرف عليها هذه المديرية.

وغالباً يتم تنشيط هذه الطرق (تنفيذ هذه البرامج) نتيجة لمؤثر خارجي، أي وصول رسالة من خارج الكيان تطلب تنفيذ هذه العملية أو تلك . وهكذا فإن الكيانات تتفاعل مع بعضها البعض (أي مع بيئتها) من خلال الرسائل ،

حيث تنشط كل رسالة طريقة ما في الكيان ، تؤدي كما ذكرنا إلى تغيير قيم خصائص ذلك الكيان .

3. الكيانات والأصناف (الأنواع) Object and Classes

يعرف النوع أو الصنف Class بأنه كيان يستخدم لوصف مجموعة كيانات لها نفس السمات أو الملامح Features، ويقصد بهذه الملامح خصائص معينة للنوع تميزه عن الأنواع الأخرى . كما يمكن القول أن للكيان الذي ينتمي إلى نوع معين ملامح أخرى بالإضافة إلى خصائصه وطرقه الخاصة به . ولا يتضمن المدخل الموجه للكيانات أية مجموعة قياسية لهذه الملامح ، فالكيانات يمكن أن تمتلك أي عدد من الملامح المختلفة . وبصورة عامة فإن الكيانات يمكن أن يكون لها الملامح التالية :

- حالات الكيان التي يمكن أن تنشط الطرق المختلفة فيه.
- القيود المفروضة بين صفات أو خصائص الكيان.
- التدقيقات القبلية والبعدية للحالات المختلفة للتأكد من الرسائل والاستجابات.
- الأحداث Triggers التي تنشط الرسائل في حالة أو ظرف معين .

من الخصائص الهامة التي تميز هذا الأسلوب إمكانية نمذجة النوع Object Class ككيان ، بنفس الطريقة التي يتم بها نمذجة الكيانات المنتمية لذلك النوع. ولذلك نجد مثلا نوع كيانات اسمه " المديريات " ، وكل مديرية فيه هي عبارة عن حالة حدوث لهذا النوع . ومن الممكن عند التنفيذ ، أن يتم تخزين هذا النوع المسمى " المديرية " ككيان مستقل، وكذلك تخزين كل مديرية ككيان مستقل

أيضاً، ويتم تخزين الطرق الموجودة في الكيان مرة واحدة في كيان النوع لكي لا يتم تكرارها في الكيانات الخاصة بكل مديرية .

4. نمذجة الكيانات :

يمكن نمذجة الكيانات باستخدام الطريقة البيانية وبشكل مشابه نوعاً ما للمخططات المستخدمة سابقاً . ويتم تمثيل الكيان عادة بشكل مستطيل مقسم إلى أجزاء كما هو مبين في الشكل (3.14) .

اسم الكيان Object name
الخصائص Properties
الطرق Methods
القيود Constraints

شكل (3.14) التمثيل البياني للكيان .

يتضح من الشكل (3.14) أن الكيان يتم تمثيله بيانياً بشكل مستطيل مؤلف من أربع أقسام وتتضمن المعلومات التالية :

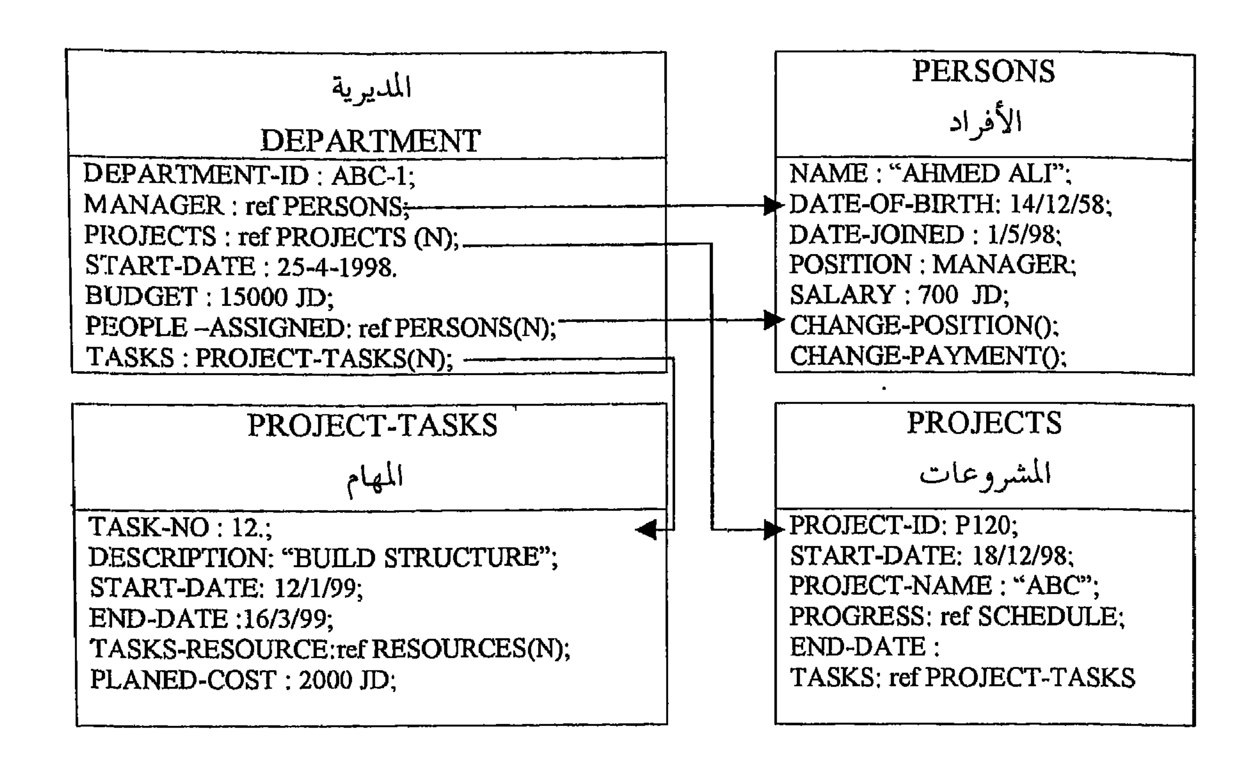
- اسم الكيان .
- خصائص الكيان .
- الطرق الخاصة بالكيان .
- القيود .

فمثلاً يمكن استخدام هذه الطريقة لتمثيل الكيان الموجود في الشكل (2.14) على النحو التالي : .

DEPARTMENT	المديرية
DEPARTMENT-ID : ABC; MANAGER : ref. PERSONS; PROJECTS : ref. PROJECTS(N); START-DATE: 25-4-1998. BUDGET : 24000 JD; PEOPLE –ASSIGNED : ref. PERSONS(N); TASKS : ref. DEP-TASKS(N);	الخصائص
ADD-PERSON( ); DEL-PERSON( ); CHANGE-BUDGET( ); CREATE-TASK( ); ADD-PROJECT( ); CLOSE-PROJECT();	الطرق
ENSURE number(PEOPLE-ASSIGNED) < 200 number(PROJECTS) ≤ 6	القيود

الشكل (14.4) طريقة نمذجة الكيانات (مثال)

ويتضمن النموذج عادة عدداً من الكيانات المرتبطة مع بعضـــــها البعــض بعلاقات محددة. ويتم إنشاء هذه العلاقات من خــــلال المراجــع References أي الاشارات المرجعية المختلفة. والإرجاع إلى كيان ما يتم عادة من خلال خصـــائص الكيان .ويتم تمثيل هذه الإشارات المرجعية في النماذج البيانية بشكل خطوط تصـــل بين الكيانات كما هو مبين في الشكل (5.14) أدناه :

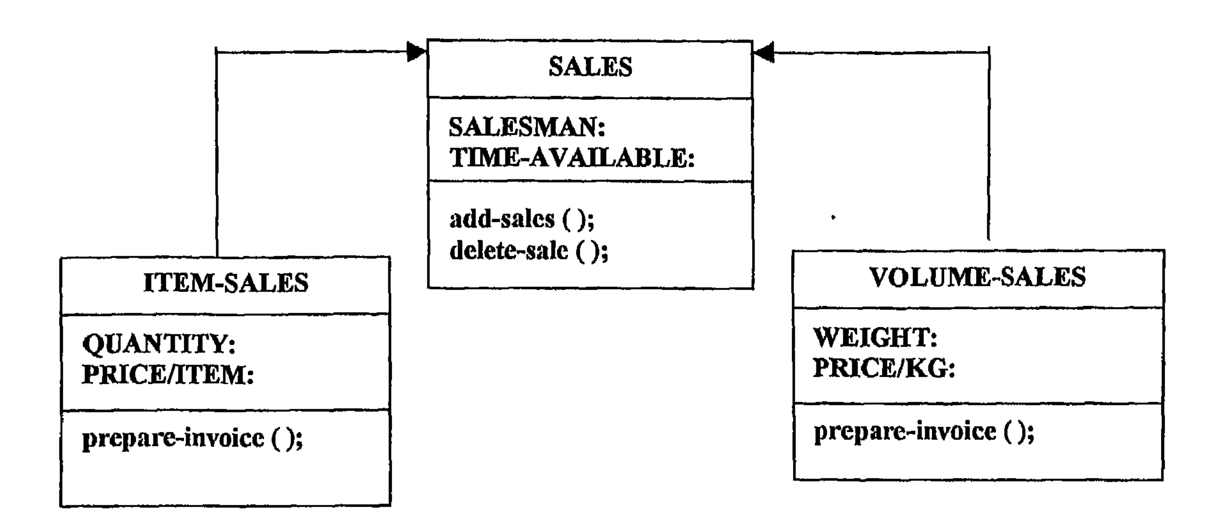


شكل (5.14) نموذج لتركيب كيانات النظام .

5. مفهوم التوارث Inheritance :

يقصد بالتوارث Inheritance استخدام نفس السمات أو الملامح الموجودة لدى كيان آخر . ويشبه هذا المفهوم إلى حد كبير الفكرة المستخدمة لنمذجة المجموعات الفرعية Subsets والتي تعرفنا إليها في الفصل الثامن . ووفقاً لهذا المفهوم يمكن لكائن ما أن يرث سماته أو خصائصه من كيان آخر . كما يمكن للكيان أن يكون له بعض الملامح الإضافية . أو إذا دعت الحاجة ، استبدال بعضاً من ملامحه (سماته). ويبين الشكل (6.14) بنية توارث بسيطة ، تتضمن نوعين من عروض البيع ، الأول يستند إلى فقرات محددة والآخر إلى كميات الفقرات . وكلا هذين

النوعين من المبيعات يرثان الملامح الموجودة في الكيان Sales بالإضافة إلى الملامح الخاصة بكل منهما :



شكل (6.14) بنية التوارث

وهكذا فإن لكل من الكيانات ITEM-SALES و VOLUM-SALES ملامحهما الخاصة بالإضافة إلى الملامح التي يرثوها من الكيان الذي يمثل النوع SALES .

وينبثق عن مفهوم التوارث مفهوماً آخر هو **التخصص** Specialization ، ويقصد به أن يكون للكيان صفات أخرى غير تلك يرثها عن كيان آخر . وهكذا بالرجوع إلى المثال أعلاه يمكن أن نقول بأن الكيانات ITEM-SALES و VOLUME-SALES هما تخصصان للكيان SALES فهما يرثان جميع سماته وملامحه ويتخصصان بسمات إضافية خاصة بهما. ويعتبر مفهوم التوارث مهم جداً لأنه يوفر للمصمم إمكانية تخصيص الكيانات دون الحاجة إلى تغيير الرسائل Calling

Messages . فمثلاً يمكن إضافة نوع جديد من المبيعات وليكن اسمه -DAILY SALES، ككيان متخصص ثالث للكيان الأصلي SALES ويرث عنه جميع خصائصه وعملياته، بالإضافة إلى سماته الخاصة به.

وعند دراسة التوارث لا بد من التعرف على مفهوم أخر ذو علاقة وثيقة هو **تعددية الأشكال** Polymorphism، وهو يعني قدرة بنية معينة على اتخاذ عدد من الأشكال . وفي المدخل الموجه للكيانات OOP ، ينطبق ذلك بشكل خاص على قدرة الرسالة تغيير تأثيرها تبعاً للحدوث Instance في الكيان المستدعى called object فمثلاً الرسالة prepare-invoice() المرسلة من الكيان SALES سوف تختار الطريقة المناسبة تبعاً لنوع المبيعات الذي ستتعامل معه . وأكثر من ذلك فإنه عند إضافة كيان متخصص جديد يمتلك مواصفات جديدة لهذه الرسالة فإنها سوف تختار الطريقة المناسبة إذا وجهت إلى هذا الكيان الجديد .

وأخيراً تجدر الإشارة إلى أن التوارث يمكن أن يكون متعدداً Multiple Inheritance ، حيث يمكن للكيانات أن ترث سمات وملامح من عدد من الكيانات الأخرى (من اكثر من كيان واحد) .

6. خطوات تطبيق أسلوب التحليل والتصميم الموجه نحو الكيانات

يمكن تطبيق هذا المدخل الجديد لتطوير أنظمة المعلومات ، والذي يقوم كما ذكرنا على مفهوم الكيانات باعتبارها تمثل العناصر الرئيسة المكونة للنظام ، من خلال الخطوات التالية :

1.6- تحديد الكيانات وعملياتها وخصائصها :

تهتم هذه الخطوة الأولى بتعريف الكينونات التي يتكون منها النظام . ولقد عرفنا الكيان بأنه أي شيء Object في النظام يمكن وصفه من خلال مجموعة من البيانات Set of Data وسلسلة من العمليات Set of actions . فبينما تصف البيانات خصائص attributes الكيان ، تمثل العملية حدث معين يمكن أن يتعرض له في لحظة زمنية محددة ، فمثلاً عندما يقدم الزبون طلباً لشراء منتجات معينة ، أو عندما يقوم المتعامل بتقديم طلب للحصول على قرض من البنك ، أو عندما يقدم الطالب طلباً للتسجيل على مقررات معينة للفصل القادم وهكذا .

يتم وصف الكيانات (حالة الكيانات) من خلال ما يسمى بخصائص الكيان Object attributes. وعندما تتغير هذه الخصائص ينتقل الكيان من حالة إلى حالة أخرى . ويتم تغيير هذه الخصائص كما ذكرنا أعلاه ، من خلال العمليات actions الموجودة في هذا الكيان. وكذلك الحال مع العمليات حيث يتم توصيفها من خلال مجموعة الخصائص التي تمثلها action attributes . فمثلاً عملية التسجيل على مقررات الفصل الدراسي القادم (تقديم طلب تسجيل مقررات للفصل القادم) يتم وصفها من خلال مجموعة من البيانات مثل : اسم الطالب ورقمه وعنوانه، وتاريخ الطلب وأرقام وأسماء المقررات التي يرغب بدراستها وكذلك الشعب الدراسية لهذه المقررات التي يرغب الالتحاق بها وهكذا .

يتم تعريف الكيانات وخصائصها وعملياتها في هذه الخطوة من خلال القيام بالمهام التالية :

أ) **تحديد الكيانات المكونة للنظام الذي تجري دراسته** ، أي الأشياء أو الكيانات التي يتعامل معها . فمثلاً عند دراسة نظام الحجز في شركة

نقل الركاب ، فان الخطوة الأولى هذه تتضمن تحديد الكيانات المكونة لهذا النظام . وفي هذه الحالة فإن الكيانات التي تمثل نظام الحجز هذا يمكن أن تكون الزبون والرحلة والمقعد أو المقاعد التي يحجزها الزبون.

ب) **تحديد الخصائص (البيانات) المرتبطة بكل كيان** : وهذا يعني إعداد قائمة بالخصائص التي يمكن من خلالها وصف كيانات النظام ففي مثالنا أعلاه يمكن وصف الزبون باستخدام الخصائص التالية : الاسم – العنوان – رقم الهاتف .أما الرحلة فيمكن وصفها من خلال البيانات التالية : رقم الرحلة – رقم الحافلة (الباص) – الوجهة (المدينة) – موعد الرحلة وهكذا .

جـ) **دراسة الأحداث التي يمكن أن تقع في بيئة النظام وتحديد تأثيرها على الكيانات المكونة له** : ففي هذه الخطوة يتم البحث عن الأحداث الخارجية التي يمكن أن تقع في بيئة النظام ، وإعداد قائمة بهذه الأحداث، ثم دراسة تأثير كل منها على الكيانات المختلفة للنظام . فمثلاً يمكن للزبون أن يتصل بالشركة ليطلب إلغاء حجزه على رحلة ما، أو تغييره من رحلة إلى أخرى . أو إجراء حجز جديد يتضمن مقعداً أو عدداً من المقاعد .

ولا تقتصر الأحداث الخارجية على تلك الأحداث التي يمكن أن يقوم بها الزبائن ، بل تشمل أيضا ما يمكن أن تقوم به إدارة الشركة . فمثلاً إلغاء رحلة معينة ، أو تأجيلها أو استبدال حافلة بأخرى يختلف عدد

مقاعدها عن الحافلة المخصصة لهذه الرحلة . وكذلك تغيير أسعار الرحلات وغيرها .

د) **تحديد العمليات actions اللازمة للإستجابة لهذه الأحداث** : وهنا يجب تحديد الاستجابة إلى هذه الأحداث ، فمثلا عندما يقوم الزبون بإلغاء الحجز يجب تزويد النظام ببيانات مثل : اسم الزبون – رقم الرحلة – رقم أو أرقام المقاعد التي يرغب إلغاء حجزها وهكذا .

هـ) **تحديد الخصائص المرتبطة بكل عملية من العمليات** : حيث يتم إعداد قائمة بالخصائص المتعلقة بكل عملية من العمليات اللازمة للاستجابة على الأحداث الخارجية للنظام . فمثلا عملية الحجز يمكن توصيفها من خلال البيانات التالية : اسم الزبون – العنوان – رقم الرحلة – رقم المقعد أو المقاعد – الدرجة (إذا كانت الحافلة مقسمة إلى درجات) وهكذا .

وأخيرا في نهاية هذه الخطوة يقوم المحلل بإعداد قائمة أو جدول بالكيانات والعمليات وخصائصها تمهيدا للخطوة التالية والمتعلقة بتحديد الترتيب الزمني للأحداث والعمليات المتعلقة بكل كيان .

2.6– تحديد الترتيب أو التسلسل الزمني للعمليات في النظام :

لقد ذكرنا سابقا أن الأحداث تقع في لحظات زمنية وبتسلسل معين . وقد تؤدي بعض الأحداث إلى وقوع أحداث أخرى (أي يتبعها وقوع أحداث أخرى). كما أن بعض الأحداث لا يمكن أن تقع إلا إذا سبقها وقوع أحداث معينة وهكذا . وتهتم هذه الخطوة بتحديد التسلسل الزمني لوقوع الأحداث (التسلسل الزمني

للعمليات المختلفة في النظام) ، وهذا يعني تحديد علاقة عمليات النظام ببعضها البعض . ومن أمثلة هذه العلاقات ما يلي :

- عملية الحجز يجب أن تسبق جميع العمليات الأخرى في النظام .
- عملية إلغاء حجز مقعد ما يجب أن تسبقها عملية حجز ذلك المقعد وهكذا .

وتستخدم مخططات تاريخ حياة الكينونة Entity Life History التي تمت دراستها في الفصل الثامن لتحديد التسلسل الزمني للأحداث (العمليات) الخاصة بالكيانات المختلفة للنظام .

3.6 – تنفيذ النظام :

تتضمن هذه الخطوة المهام التالية :

- توصيف بيانات (خصائص) الكيانات والعمليات باستخدام لغة البرمجة التي سيتم بواسطتها كتابة برامج النظام.
- كتابة برامج العمليات المرتبطة بكل كيان.
- ربط العمليات والبيانات الخاصة بكل كيان في إطار البرنامج الخاص به Object Program.
- ربط برامج الكيانات معا في البرنامج الكلي للنظام.

7. نمذجة الكيانات ودورة حياة تطوير النظام :

بعد أن تعرفنا إلى المفاهيم الأساسية للتحليل والتصميم الموجهة للكيانات، سنتعرف على الطرق المختلفة التي يمكن من خلالها استخدام هذا الأسلوب خلال دورة حياة تطوير الأنظمة المعلوماتية. وبشكل عام يمكن الاستفادة من هذا المدخل الجديد في نمذجة الأنظمة بطرق عديدة أهمها :

1- نمذجة الكيانات انطلاقا من مخطط الكينونة – العلاقة :

وذلك بتحويل كل مجموعة كينونة entity set إلى كيان Object ، فتكون خصائص هذه الكينونة attributes هي عبارة عن صفات Properties هذا الكيان ، بعد ذلك يمكن إضافة الطرق الخاصة بهذا الكيان وتحديد السلوك العام له ، أي التفاعلات Interactions التي يمكن أن تحدث بينه وبين الكيانات الأخرى .

2- بناء نموذج الكيانات بشكل مباشر :

في البداية لم تكن لدى المحللين الطرق المناسبة لضمان اكتمال تحليل وتصميم نماذج الكيانات . لذلك كانوا يقومون بسرد جميع الخدمات أو العمليات التي يمكن أن يقوم بها الكيان . مع ذلك كان من الصعب على المحلل (المصمم) التأكد من أنه قد حدد تماما جميع الطرق أو الخدمات المرتبطة بالكيان . ولذلك اتجه البحث عن طرق تحليل جديدة يمكن بواسطتها تحديد جميع مكونات الكيان بدقة . ولقد تم اقتراح عدة منهجيات في هذا الموضوع ، تعتمد في مجملها أحد المدخلات التالية :

- إعداد مخطط الكيانات Objects Diagram يمثل جميع الكيانات وخصائصها وهو يشبه إلى حد كبير مخططات العلاقة –. الكينونة (E-RD).
- إعداد نموذج ديناميكي يمثل الحالات التي يمكن أن يمر بها الكيان . ويشبه هذا النموذج إلى حد كبير مخطط دورة حياة الكينونة الذي تمت دراسته في الفصل الثامن . يعرض هذا المخطط الحالات التي يمكن أن يمر بها الكيان خلال حياته والتحولات التي يمكن أن تطرأ عليه . وتغيير الحالات يعني تغيير قيم خصائص الكيان . وهذه التحولات يمكن تنشيطها

إما من خلال حدث داخلي في الكيان Internal event، أو من خــــلال رسالة تأتي من كيان آخر . وكل تحول من هذه التحولات يتطلب طريقة method لكي يتم تغيير قيم خصائص الكيان وفقا لما يتطلبــه الحــدث . وهكذا فإن كل تحويل Transaction سيمثل طريقة من الطرق الموجــودة في هذا الكيان .

- استخدام النموذج الوظيفي Functional model الذي يبين التحولات Transformations التي يمكن أن تحدث لخصائص الكيان . وكل من هــذه التحولات يكون سببه حدث ما ، وتتم نمذجته بطريقة مشابهة للطريقــة المستخدمة في مخططات التدفق DFD .

وأخيرا يمكن استخدام هذا المدخل في عدة أماكن من دورة حياة تطويــــر الأنظمة منها :

- استخدام مفاهيم هذا التوجه خلال تنفيذ النظام : وهذا يعني أن تتــم عملية التحليل بالطريقة التقليدية (مخططات تدفق البيانات ومخططــات الكينونة – العلاقة وغيرها) ، ثم تنفيذ النظام الجديد بكتابة الــبرامج بإحدى لغات البرمجة الموجهة للكيانات OOP Language
- الشروع منذ بداية دورة حياة تطوير النظام باستخدام المدخل الموجــه للكيانات في تحليل ونمذجة النظام، ثم تنفيذ النظام بالطرق التقليدية.
- استخدام التحليل الموجه للكيانات ثم ترجمته أي تنفيذه بلغات البرمجــة وأنظمة إدارة قواعد البيانات الموجهة للكيانات.
- استخدام هذا المدخل الموجه للكيانات لدعم مدخل التطويــر الــذي يقوم على تركيب النظام System Development مـــن كيانــات تم

تطويرها مسبقا. وهذا يتطلب تصميم الكيانات بحيث يتوفر فيها إمكانية ربطها بسهولة مع غيرها من الكيانات الأخرى في النظام.

- استخدام المدخل الموجه للكيانات في التطوير التدريجي للنظام، حيث يمكن في البداية تطوير عدة كيانات وربطها معا واختبارها، ثم إضافة كيانات أخرى إليها. كما يمكن تخصيص هذه الكيانات وتخزينها في مكتبة الكيانات بهدف إعادة استخدامها. وتتجه منهجيات التطوير الحديثة إلى دمج أسلوبي التطوير التدريجي والتركيب معا، للاستفادة من مزاياهما، وهو اكتساب الخبرة وبناء النظام بشكل تدريجي من خلال تطوير كياناته واختبارها أولا بأول وتخزينها وإعادة استخدامها عند بناء الأنظمة الجديدة.

8. مكتبات الكيانات Objects Libraries

يقصد بالمكتبة مجموعة من الكيانات التي يمكن الاستفادة منها واستخدامها لتطوير أنظمة المعلومات الجديدة في المنظمة. وهكذا فإنه يمكن بناء مكتبة تضم الكيانات التي يمكن إعادة استخدامها، وتوفير الطرق المناسبة لتركيب هذه الكيانات أي ضمها إلى الأشياء التي يجري تطويرها. وعادة يتم بناء مثل هذه المكتبات بطرق مختلفة منها:

- عند بناء الكيانات في مرحلة تنفيذ النظام يمكن الاحتفاظ بنسخ منها في المكتبة بعد أن يتم تصنيفها لتسهيل الرجوع إليها، واستخدامها من قبل المستفيدين المحتملين (المبرمجين) الذين قد يحتاجون إليها في عملهم سواء في هذا النظام أو في الأنظمة الأخرى. وعندما يحتاج أحد المبرمجين لهذه

الكيانات يمكن أن يستخدمها كما هي بشكلها الحالي، أو يجري عليها بعض التعديلات لتلبية احتياجات محددة.

- تطوير كيانات ذات طبيعة عامة يطلق عليها Skeleton Objects، أي كيانات ذات وظائف عامة ووضعها في المكتبة، وتكون هذه الكيانات مصنفة حسب الوظائف التي تقوم بها، ويقوم المبرمجون بإعادة استخدام هذه الكيانات بعد أن يخضعوها لعملية موائمة Customization لتتناسب مع طبيعة التطبيقات والنظم التي يقومون بتطويرها.

أسئلة الفصل

1- اشرح المفاهيم الأساسية لمدخل تطوير الأنظمة الموجه للكيانات؟

2- اشرح بنية الكيانات وكيفية تمثيلها؟

3- اشرح مفاهيم التوارث والتخصص؟

4- ما المقصود بمفهوم تعددية الأشكال في التطوير الموجه للكيانات؟

5- كيف يتم ربط الكيانات مع بعضها البعض في إطار النظام؟

6- عدد واشرح خطوات تطبيق أسلوب التحليل والتصميم الموجه للكيانات؟

7- ما هي المهام التي يتم من خلالها تعريف الكيانات وتحديد خصائصها وعملياتها؟

8- ما هي الطرق التي يمكن من خلالها استخدام مدخل التطوير الموجه للكيانات في دورة حياة تطوير الأنظمة؟

9- ما هي مكتبات الكيانات وكيف يمكن بناءها؟

الفصل الخامس عشر

حوسبة عمليات التحليل والتصميم

لقد تعرفنا خلال دراسة الفصول السابقة إلى أهم الأدوات المستخدمة في عمليات التحليل والتصميم . فبواسطة هذه الأدوات يمكن إنجاز المهام المختلفة بالدقة والجودة والكفاءة المطلوبة. ففي النمذجة تستخدم مخططات تدفق البيانات (DFDs) ومخططات هياكل البيانات (E-RD). وفي بناء النماذج التجريبية تستخدم لغات الجيل الرابع كمولدات الشاشات والتقارير والتطبيقات وقواميس البيانات ومكتبات البرامج الجاهزة وغيرها .

ومع تطور وإنشاء التطبيقات الحاسوبية المختلفة بدأ التفكير بالاستفادة من الحاسوب كأداة لمساعدة فريق التطوير من محللين ومصممين في إنجاز عمليات النمذجة والتوثيق والعديد من المهام المتعلقة بالتحليل والتصميم، بهدف زيادة الإنتاجية وتقصير الزمن اللازم لتطوير الأنظمة إلى أدنى حد ممكن .

ولقد ظهرت خلال السنوات الماضية العديد من البرمجيات الجاهزة المصممة خصيصاً للمساعدة في أتمتة عمليات التحليل والتصميم، ويطلق حالياً على هذه البرمجيات تسمية **أدوات هندسة البرمجيات بمساعدة الحاسوب**، Computer - Aided Software Enginerering (CASE) Tools. كما ظهرت أدوات جديدة اكثر تطوراً أطلق عليها **أدوات هندسة المعلومات** Information Engineering Tools (IE). وسنتعرف في هذا الفصل إلى هذه الأدوات وما توفره من مزايا

وتسهيلات ودعم لعمليات التطوير بعامة ، والتحليل والتصميم بخاصة ، والوظائف التي تتيحها في هذا المجال .

1. أهمية ودور أدوات هندسة البرمجيات بمساعدة الحاسوب :

تهدف هندسة البرمجيات إلى المساعدة في إنتاج برمجيات ذات **جودة عالية وبتكاليف معقولة وبأقل زمن ممكن** . وتعتبر أدوات CASE و هندسة المعلومات مهمة وضرورية في إنجاز عمليات التحليل والتصميم والتنفيذ، أي يمكن استخدامها خلال جميع مراحل تطوير النظام . فهي تساعد في تحسين إنتاجية هذه العمليات بشكل ملحوظ يصل أحيانا إلى عدة أضعاف .

كما تساعد هذه الأدوات في تطوير أنظمة معلومات ذات جودة عالية، فمن خلالها يتاح لمطوري الإنتاجية استخدام إمكانات المعالجة الحاسوبية وتوظيفها لمساعدتهم في إنجاز عمليات التطوير بدءً من تجميع المعلومات والحقائق عن النظام الحالي ، وانتهاءً باختبار الأنظمة الجديدة وتشغيلها وصيانتها .

ويمكن تلخيص إنجاز مزايا هذه الأدوات على النحو التالي :

1.1 زيادة إنتاجية عملية التطوير : Improve Productivity

يساعد استخدام أدوات هندسة البرمجيات والمعلومات في تسريع عمليات التطوير، حيث يمكن تنفيذ هذه العمليات بوقت اقل من الوقت اللازم لإنجازها بالطرق التقليدية ، مما ينعكس بشكل عام في زيادة الإنتاجية. أي إنتاج أنظمة وتطبيقات اكثر من قبل فريق التطوير، ويعود ذلك بالطبع إلى استخدام الحاسوب وإمكاناته الكبيرة في أداء مهام وأنشطة التطوير .

فمثلاً باستخدام هذه الأدوات يمكن رسم مخططات تدفق البيانات ونمذجة بيانات النظام وإعداد قاموس النظام ونمذجة النظام بسرعة كبيره بالمقارنة مع الطريقة اليدوية لإنجاز هذه العمليات . وهكذا فانه بفضل استخدام هذه الأدوات الحاسوبية تصبح دورة التطوير قصيرة جداً (أيام أو أسابيع بدل الأشهر والسنوات)، وكذلك فان الأنظمة المصممة باستخدام هذه الأدوات تكون ذات دقة اكبر.

2.1 تحسين الفعالية Improve effectiveness

يقصد بالفعالية Effectiveness إختيار الإجراءات أو الطرق المثلى لتحقيق الهدف المنشود . حيث توفر أدوات هندسة البرمجيات بمساعدة الحاسوب، وأدوات هندسة المعلومات لمطوري الأنظمة إمكانات متعددة لإنجاز الأعمال المطلوبة بالطريقة الصحيحة ، وتعفيهم إلى العمليات الروتينية المرهقة للفكر أو المملة. فمثلاً تساعد أدوات الرسم البياني لمخططات التدفق المحللين في إعداد نماذج النظام، وتجعلهم يركزون تفكيرهم في تحديد متطلبات النظام واحتياجات المستخدمين ، بدلاً من إضاعة الوقت في إعداد الرسوم والنماذج البيانية المختلفة .

وهكذا فإن أهمية استخدام هذه الأدوات تتجلى أيضا في توظيف إمكانات الحاسوب **لأتمتة العمليات الروتينية** وتكريس الوقت الكافي لتجميع المعلومات ودراستها وتحليلها ومناقشتها مع أعضاء فريق التطوير . وبفضل تطوير هذه الأدوات المحوسبة أصبح بالإمكان إعداد الرسوم والمخططات اللازمة لعملية التطوير بسرعة كبيرة وبجودة عالية ، والاهم من ذلك توفرت إمكانية تعديلها وإدخال التغييرات المختلفة فيها بسرعة أيضا ، كما توفر هذه الأنظمة إمكانات تحويل البيانات بين أدوات التحليل والتصميم المختلفة ، فمثلاً يمكن تحويل مخططات تدفق البيانات إلى مخططات هيكلية للبرامج ، وكذلك تحويل مخططات الكينونة – العلاقة إلى مخطط

تدفق بيانات وبالعكس. مما يتيح لفريق التطوير إمكانات واسعة ومرونة كبيرة في إعداد وثائق النظام . وبالإضافة إلى ذلك تتيح أدوات هندسة البرمجيات إمكانية مطابقة وتدقيق المعلومات الموجودة في مخططات تدفق البيانات مع المعلومات الموجودة في مخططات البيانات وفي قاموس البيانات وتوصيف العمليات ومع المخططات الهيكلية ، مما يضمن الدقة والاكتمال في هذه النماذج .

3.1 تحسين الجودة Improve Quality :

إن استخدام إمكانات الحوسبة في إنجاز مهام وعمليات تحليل وتصميم أنظمة المعلومات سيؤدي حتما إلى إعداد أنظمة تتميز بجودة عالية . فبفضل التقسيم الأمثل للعمل بين أعضاء فريق التطوير والحاسوب يتمكن أعضاء الفريق من تركيز اهتمامهم على تحديد احتياجات المستخدم وتوصيف متطلبات النظام ، وابتكار الحلول التصميمية المختلفة ، وهذا يحتاج إلى إمكانات إبداعية خلاقة ولذلك تبقى مسؤولية أعضاء الفريق .

أما العمليات الروتينية كتسجيل المعلومات وإعداد الرسوم والمخططات كأداة قاموس البيانات وتوصيف أشكال الشاشات والتقارير وإعادة رسمها وتوصيفها بعد كل تعديل، فيمكن القيام به بصورة أدق وافضل بواسطة الحاسوب. وهذا يسمح للمطورين بتركيز اهتمامهم في التفكير بالحلول الجديدة وتوليدها، ويجعل هذه الأنظمة جديدة فعلا ، أي توفر إمكانات ووظائف جديدة للمستخدم، وليست عبارة عن نسخ محدثة للأنظمة الحالية. كما تساعد أدوات هندسة البرمجيات في التطبيق التلقائي للمعايير القياسية المتعلقة بعملية تطوير الأنظمة المعلوماتية في المنظمة. وتتعلق هذه المعايير غالبا بطرق توصيف البيانات (التسميات - الحجوم - أنواع البيانات وغيرها) والعمليات، وكذلك طريقة صياغة أشكال الشاشات

والتقارير (تصميم الشاشات والتقارير) أو طريقة استخدام المفاتيح الوظيفية في الحواسب الشخصية وغيرها .

وبفضل استخدام أدوات هندسة البرمجيات بمساعدة الحاسوب يصبح بالإمكان تعريف وإدخال هذه المعايير إلى الحاسوب ، ليتم تطبيقها بصورة تلقائية عند إنجاز مختلف مهام التحليل والتصميم . فمثلاً عند تصميم أي شاشة أو تقرير تطبق بصورة تلقائية المعايير المعتمدة لهذا الغرض، والتي تحدد شكل الشاشة أو التقرير وأسلوب توزيع المعلومات المختلفة فيه. ومن الواضح إن هذا التطبيق التلقائي للمعايير القياسية يضمن دقة وجودة التصاميم وبالتالي جودة ودقة البرمجيات الناتجة عن عملية التطوير، ويحقق الانسجامية الكاملة بين مختلف مكونات نظام المعلومات.

4.1 توفير مخزون عام لمعلومات ولوثائق النظام :System Repository

تتراكم لدى فريق التطوير خلال مراحل تطوير النظام مجموعة كبيرة من الوثائق والمعلومات المتعلقة بالنظام الحالي ومشاكله ونقاط الضعف فيه ، ثم متطلبات تطويره وكذلك وثائق التحليل المختلفة ووثائق التصميم والتصاميم البديلة وغيرها الكثير من المعلومات والوثائق المستخدمة خلال عملية التطوير . ومثل هذه المعلومات قد تكون مفيدة عند تطوير الأنظمة الأخرى في المنظمة ، ويمكن الرجوع إليها في المستقبل سواء عند صيانة النظام أو عند تطوير أنظمة المعلومات الأخرى .

ومن المزايا الهامة لاستخدام أدوات هندسة البرمجيات بمساعدة الحاسوب كونها توفر مخزناً آليا لجميع الوثائق والمعلومات المتعلقة بتفاصيل النظام ، وتتيح إمكانية استرجاعها والوصول إليها بسرعة عند الحاجة إليها . وهذا يعني المحافظة على الجهود المبذولة والاستفادة منها ، وعدم تكرار المهام واختراع العجلة من جديد

عند تطوير كل نظام . مما يعني التوفير في الوقـــت والجـــهود وكذلـــك ضمـــان الانسجامية والاتساق بين أنظمة المعلومات المختلفة.

5.1 الاستفادة من إمكانات الحاسوب :

يتيح استخدام أدوات هندسة البرمجيات بمساعدة الحاسوب وكذلـك أدوات هندسة المعلومات ، فرصة الاستفادة من الإمكانات الكبيرة التي توفرها الحواسيب في إنجاز مهام التحليل والتصميم المختلفة مثل :

- إعداد رسوم وجداول ومخططات بيانية ذات دقـــة عاليـــة وباســـتخدام الألوان، وإمكانية التكبير والتصغير وتغيير مقيـــاس الرســـم ، واســـتخدام خطوط مختلفة وقياسات مختلفة وغير ذلك من الإمكانات الكثـــيرة الـــتي يوفرها الحاسوب.
- إمكانية التدقيق الآلي (التلقائي) للأخطـــاء والتـــأكد مـــن التكـــامل Completeness والتناسق Consistency بين مختلف وثائق النظام ، وكذلك تدقيق المواصفات واكتشاف النقاط المهملة أو المحذوفة .
- إمكانية التنقل بين مختلف وثائق النظام وإظهار عدة وثائق على الشاشة في آن معاً .
- سهولة إعداد وثائق ومخططات النظام ، وكذلك سهولة تعديلها وصيانتها وتحديثها .كما إن إدخال أي تعديل في وثيقة ما، يؤدي بصورة تلقائية إلى تعديل جميع الوثائق الأخرى ذات العلاقة بهذه الوثيقة . فمثلاً تغير تســـمية تدفق بيانات معين في مخطط تدفق البيانات ، يؤدي أيضا إلى تعديل تســـمية هذا التدفق في قاموس البيانات وفي وثائق توصيف العمليات وهكذا .

- توفير إمكانية تحديد العلاقات المتبادلة بين مختلف وثائق النظـــام. فمثـــلا يمكن التعرف على البرامج التي تستخدم ملف معين، أو حقل بيانات مـــا. كما يمكن تحديد الملفات والبرامج التي يمكن أن تتأثر نتيجة لتغيير توصيـــف حقل بيانات معين، وهكذا.

2. تصنيف أدوات هندسة البرمجيات بمساعدة الحاسوب:

يمكن تصنيف أدوات هندسة البرمجيات بمســـاعدة الحاســـوب، بحســـب مراحـــل عملية التطوير التي تستخدم فيها هذه الأدوات والأنظمة إلى ثلاث فئـــات هي :

- أدوات مرحلتي التحليل والتصميم.
- أدوات مرحلتي التنفيذ والصيانة
- أدوات متكاملة لجميع مراحل عملية التطوير .

وفيما يلي عرض لأهم خصائص واستخدامات كل فئـــة والأدوات الـــتي تتضمنها :

1.2 أدوات مرحلتي التحليل والتصميم :

تسمى هذه الأدوات أيضاً " أدوات المراحل الأولى " Front - End CASE Tools . وهي مصممة لحوسبة الأنشطة والمهام في المراحل الأولى من عملية التطوير. وتركز هذه الأدوات على مساعدة فريق التطوير في تحليل الإحتياجـــات وإعـــداد التصاميم المنطقية للنظام الجديد . أما أهم وظائف هذه الأدوات فهي :

- **أدوات التحليل Analysis Tools** : وتتضمن أدوات رسم المخططـــات Diagramming Tools المستخدمة في رسم مخططات تدفق البيانـــات ومخططـــات

هياكل البيانات ومخططات الكينونة – العلاقة وغيرها . كما تتضمن أدوات انشاء وصيانة قاموس بيانات النظام وتوصيف العمليات . وكذلك تتضمن أدوات التحليل ما يسمى بأدوات الهندسة العكسية Reverse Engineering التي تستخدم لتحليل البرامج المصدرية للأنظمة الحالية ، وتحويلها إلى قيود في قاموس البيانات والى مخططات تدفق ونماذج بيانات .

– **أدوات التصميم Design Tools** : وتتضمن أدوات رسم مخططات البرامج (المخططات الهيكلية) وأدوات توصيف وتصميم الشاشات ووثائق الإدخال ، وكذلك أدوات توصيف وتصميم التقارير ، وأدوات توصيف ونمذجة قواعد البيانات .

– **أدوات تخزين معلومات النظام System Repository** : وتستخدم لتخزين جميع المعلومات والوثائق المتعلقة بالنظام الذي سيتم تطويره، سواء بشكله الحالي أو الجديد. وهذه الأداة هي عبارة عن مخزن عام أو قاموس مرجعي يتم فيه تخزين جميع وثائق ومعلومات النظام ، بهدف تسهيل العودة إليها واستخدامها عند الحاجة إليها.

– **أدوات التوصيف Specification Tools** : تساعد هذه الأدوات المحلل في إعداد المواصفات، أي توصيف الاحتياجات والأهداف بشكل واضح ومحدد ورسمي Formal. وكذلك تساعد في إعداد وثائق التحليل والتصميم وتلخيص نتائجها وإعداد المقترحات المختلفة لمناقشتها مع الإدارة أو أعضاء فريق التطوير . وتوفر هذه الأدوات للمحلل أو المصمم طريقة متناسقة لتوثيق المهام المختلفة وإعداد خطط العمل ومتابعة تنفيذها .

إن الميزة الرئيسية في أدوات التحليل والتصميم باستخدام الحاسوب هي أنها توفر **إمكانية التكامل** فيما بينها ، نظراً لاستخدام قاعدة تخزينية مشتركة هي **مخزن النظام** System Repository والتي أشرنا إليها أعلاه. فمثلاً عند تحديد اسم ملف أو عنصر بيانات جديد أثناء نمذجة تدفقات البيانات في النظام ، تظهر تلقائياً نافذة على الشاشة تطلب تحديد مواصفات هذا الملف أو العنصر، إذا لم تكن هذه البيانات موجودة في قاموس بيانات النظام . وكذلك الآمر عند كتابة أسماء التدفقات في نماذج توصيف العمليات يتم تدقيقها مع قاموس البيانات وإضافتها إليه في حال عدم وجودها فيه .

2.2 أدوات مرحلتي التنفيذ والصيانة :

تسمى هذه الأدوات أيضا أدوات المراحل الأخيرة Back -End CASE Tools، وهي مصممة للمساعدة في صياغة منطق البرامج وفي تحديد الخوارزميات والتوصيف المادي للبيانات ، وبناء واجهات الاستخدام وغير ذلك من المهام المتعلقة بتنفيذ النظام واختباره ووضعه قيد التشغيل . أما أهم هذه الأدوات فهي :

- **أدوات التوليد أو المولدات Generators** : وتستخدم لتوليد (إنتاج) التطبيقات Application Generators والشاشات Screen Generators والتقارير Report Generators .

ومن الواضح إن استخدام مثل هذه الأدوات يساعد بشكل ملحوظ في تسريع عمليات التنفيذ فمن خلال هذه الأدوات يمكن توصيف الإحتياجات، مما يؤدي بصورة تلقائية إلى توليد البرامج التطبيقية اللازمة لتلبية هذه الاحتياجات.

- **لغات البرمجة من الجيل الرابع (4GL)** : وهي تتميز بسهولة تعلمها واستخدامها حتى من قبل غير المختصين في البرمجة ، حيث يقوم المستخدم بتوصيف أشكال التقارير أو الشاشات المطلوبة دون الاهتمام بتفاصيل الإجراءات اللازمة للوصول إلى البيانات ومعالجتها . حيث تتم هذه المهمة من قبل الحاسوب . ولذلك يطلق على هذه اللغات تسمية " اللغات غير الإجرائية " .

وتتضمن هذه اللغات العديد من الإمكانات التي يسهل استخدامها والتي تساعد في حوسبة عمليات تنفيذ النظام كلغات الاستعلام الهيكلية SQL التي تمكن المستخدم من الوصول إلى قواعد البيانات بسرعة وسهولة .

- **أدوات الاختبار Testing Tools** : تساعد هذه الأدوات في اختبار البرامج والتأكد من صحتها من حيث الشكل والمنطق . وتقوم أيضاً بتوليد البيانات العشوائية اللازمة لعمليات الاختبار .

3.2 الأدوات المتكاملة Integrated CASE Tools :

توفر الأدوات المتكاملة مجموعة واسعة من الإمكانات الحاسوبية للمساعدة في جميع مراحل تطوير أنظمة المعلومات مما يوفر ميزة هامة وهي إمكانية بناء النظام بشكل آلي (محوسب) بناء على مخرجات مرحلتي التحليل والتصميم . أي إمكانية تحويل توصيف تصميم النظام Design Specification بشكل تلقائي إلى نظام معلومات جاهز للاستخدام . وبذلك يمكن تحقيق أتمتة شبه كاملة لمهام وأنشطة تطوير أنظمة المعلومات .

وهكذا فإن هذه الأدوات المتكاملة توفر بيئة عمل يمكن من خلالها تحويل مخرجات مرحلتي التحليل والتصميم (مخططات تدفق البيانات - المخططات الهيكلية

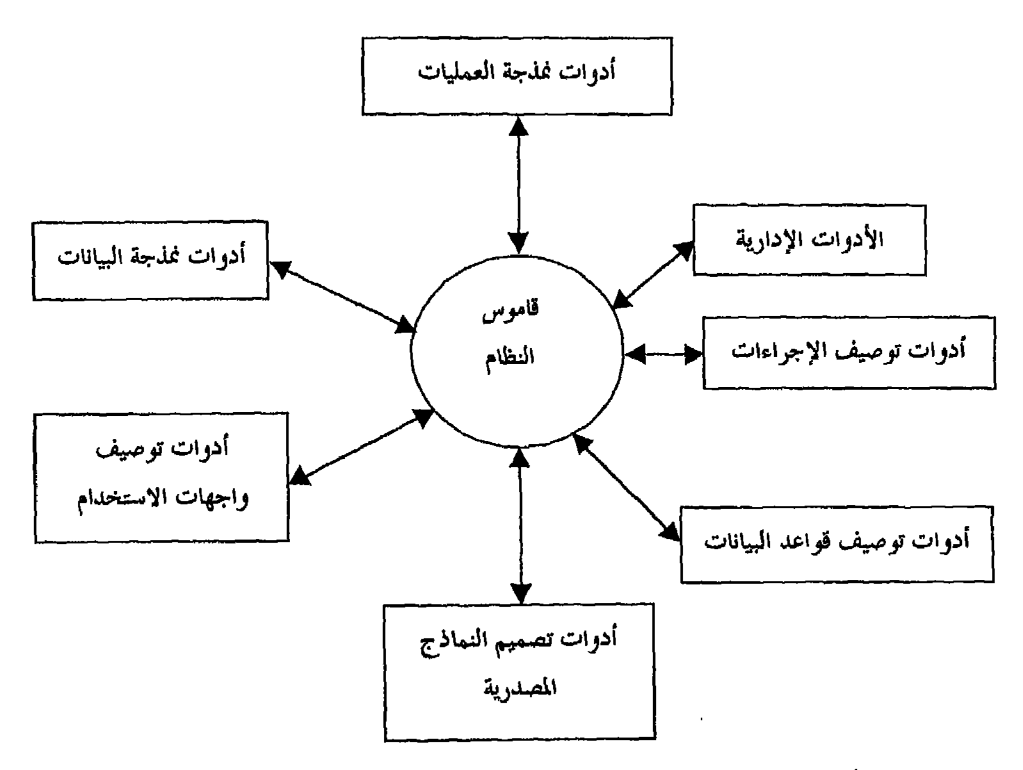
- قاموس البيانات - توصيف الوحدات الوظيفية وغيرها) إلى برامج وقاعدة بيانات النظام وواجهات استخدامه . وبالإضافة إلى توفير الإمكانات اللازمة لحوسبة أنشطة ومهام التحليل والتصميم والتنفيذ والصيانة، فإن هذه الأدوات توفر أيضاً إمكانات حوسبة عمليات التوثيق وإدارة مشاريع التطوير ومتابعتها لتتم ضمن التكلفة والزمن المحددين وبالجودة المقررة .

3. الاستخدام المتكامل لأدوات هندسة البرمجيات بمساعدة الحاسوب :

تتضمن هذه الأدوات مجموعة كبيرة من البرمجيات الجاهزة المصممة للمساعدة في إنجاز المهام المختلفة لعملية التطوير . ويمكن استخدام هذه الأدوات بشكل إفرادي أي كل أداة على حدة ، أو بشكل مجموعة من الأدوات المتكاملة ، لذلك يمكن أن نسميها صندوق الأدوات Tool Box. وبالطبع فإن الاستخدام المتكامل لهذه الأدوات يكون اكثر فائدة لما يوفره من مزايا كثيرة أهمها الربط بين مكونات نماذج النظام المختلفة . ويمكن أن يتحقق هذا الاستخدام المتكامل وفق الطرق التالية :

- **استخدام واجهه موحدة** لاستدعاء الأدوات المختلفة ، كما هو الحال في الحزمة المسمى Excelerator التي تعتمد على طريقة القوائم في التنقل بين الخيارات المختلفة واختيار الأداة.التي تناسب المهمة المطلوبة . ومن الأمثلة الأخرى في هذا المجال الحزمة المسماة Knowledge Ware التي تستخدم أسلوب النوافذ لعرض المعلومات والوثائق المختلفة في آن واحد على الشاشة . فبواسطة هذا البرنامج يمكن للمصمم أن يظهر في نفس الوقت على الشاشة كل من مخطط تدفق البيانات والمخطط الهيكلي وقاموس البيانات ومخطط الكينونة - العلاقة وغيرها.

- **استخدام مخزن معلومات مركزي** : تسمى هذه الطريقة Information Transferability حيث يمكن من خلالها نقل المعلومات التي تم تجميعها وإدخالها باستخدام أداة معينة (مخططات تدفق البيانات مثلاً) إلى أداة أخرى (مثلاً المخططات الهيكلية للمدخلات والمعالجة والمخرجات) ثم إلى أداة ثالثة كمولدات البرامج أو مولدات واجهات الاستخدام . ويلعب قاموس البيانات دوراً هاماً وأساسيا في توفير هذه الإمكانية . نظراً لان جميع أدوات هندسة البرمجيات تتفاعل مع هذا القاموس ، وتستخدم التوصيفات والتعاريف الموجودة فيه ، حيث يمثل القاموس المستودع المركزي لمعلومات النظام كما هو مبين في الشكل (1.15)



شكل (1.15) قاموس النظام هو المخزن المركزي لمعلومات النظام

وهكذا فإن المعلومات التي يتم تجميعـــها وإدخالهـــا باســـتخدام الأدوات الموجودة في CASE Tools يتم تخزينها في قاموس النظام ، وبذلك يمكن تحويلها من أداة إلى أخرى ومن مرحلة إلى أخرى حسب تقدم سير العمل في مشروع التطوير .

4. مكونات أدوات هندسة البرمجيات بمساعدة الحاسوب :

تتضمن أدوات هندسة البرمجيات بمساعدة الحاسوب بشكل عام ، المكونات الخمس التالية :

1.4 أدوات رسم المخططات Diagramming Tools :

تساعد هذه الأدوات في حوسبة مهام وأنشطة التحليل وتوثيق المتطلبـــات . وبشكل عام تتضمن الأدوات إمكانات رسم مخططات تدفق البيانـــــات (DFD) ومخططات بنية البيانات (DSD) وكذلك مخططات الكينونة – العلاقـة (E-RD) والمخططات الهيكلية Structurel Charts. وتعتبر جميعاً من الأدوات الضرورية والـتي تقوم عليها منهجية التحليل والتصميم الهيكلي . ولذلك فإن أدوات CASE تركـــز على توفير إمكانية استخدام هذه الأدوات بشكل واسع .

ولا تقتصر هذه الأدوات على توفير إمكانية رسم المخططات بل أيضاً تقـوم بتخزين تفاصيل هذه المخططات فيها ، وبالتالي فإن أي تعديل في المخطط يؤدي إلى إجراء التعديلات اللازمة في وصف تفاصيل هذه المخططات وفي الأدوات الأخرى .

وتعتبر إمكانية الرسم وتغيير الرسم أو إعادة الرسم مهمة جـــداً للمحللـــين والمصممين الذين يعتبرون هذه العمليات مجهدة ومملة وتســـتهلك معظـــم أوقـــات عملهم. ولذلك فان أدوات رسم المخططات تقدم دعماً هاماً لهم في هذا المجال .

2.4 مخزن معلومات النظام Central Information Repository :

تساعد هذه الأداة في تجميع وتحليل ومعالجة وتوزيع المعلومات المتعلقة بالنظام الذي يجري تطويره ، حيث تتم حوسبة جميع هذه العمليات بواسطة ما يسمى قاموس النظام أو مخزن معلومات النظام Information Repository . وتتضمن هذه الأداة معلومات تفصيلية عن جميع مكونات النظام ، وهذا يشمل ما يلي :

- معلومات عن إجراءات النظام ومخططاته ونماذجه .
- توصيف مخازن البيانات و السجلات وعناصر البيانات في النظام .
- توصيف عمليات النظام ووظائفه المختلفة .
- توصيف تصاميم الشاشات والتقارير والنماذج المستخدمة في النظام .

ويتم تخزين هذه المعلومات بطريقة منظمة لتسهيل الرجوع إليها واستخدامها في مراحل التطوير المختلفة . كما تتضمن هذه المخازن إجراءات رقابة وحماية تضمن دقة وإنسجامية هذه المعلومات التفصيلية المتعلقة بالنظام .

3.4 أدوات توليد واجهات الاستخدام :

يقصد بواجهات الاستخدام User Interface جميع الوسائل التي يتم من خلالها التفاعل بين الإنسان (المستخدم) والحاسوب . وهذا يشمل شاشات إدخال البيانات وقوائم الاختيار وشاشات الاستعلام والتقارير المطبوعة وغيرها .

وتقوم هذه الأدوات التي يطلق عليها تسمية مولدات الواجهات Interface Generators بتوفير الإمكانيات الحاسوبية اللازمة لتصميم الشاشات والتقارير وإنتاج النماذج التجريبية Prototypes . وتساعد هذه الأدوات كثيرا في تحسين إنتاجية عملية التطوير التي تتطلب غالبا وقتا وجهدا كبيرين . وخلال التصميم الآلي لواجهات الاستخدام (الشاشات والتقارير) تقوم هذه الأدوات بالرجوع إلى

التوصيفات الموجودة في قاموس البيانات لتحديد مواصفات الحقول التي سيتم إظهارها في الشاشات أو التقارير . كما إن العديد من هذه الأدوات توفر بيانات افتراضية يمكن استخدامها على سبيل التجربة لفحص هذه التقارير والشاشات والتعرف على مدى ملاءمتها للأهداف المرجوة منها.

4.4 توليد برامج التطبيقات Code Generators :

تقوم هذه الأدوات بحوسبة عمليات كتابة واختبار البرامج التطبيقية ، وتتضمن أساليب وتقنيات متعددة لتحويل مواصفات النظام Design Specification إلى برامج مصدرية جاهزة للتنفيذ . وتكون مولدات البرامج هذه مفيدة جدا عندما يتم ربطها بمخزن معلومات النظام ،لضمان التحديث الآلي لهذه المعلومات عند إجراء أي تعديل في البرامج المصدرية وبالعكس .

وتجدر الإشاره أخيراً إلى إن مولدات البرامج المعروفة حتى الآن ما تزال غير كاملة، أي إن البرامج الناتجة عنها تتطلب إضافة بعض التعليمات التي لا بد من إدخالها بشكل يدوي ، وخاصة فيما يتعلق بربطها مع البرامج والأحداث الأخرى في النظام .

5.4 الأدوات الإدارية Management Tools :

توفر هذه الأدوات إمكانية حوسبة الأنشطة المتعلقة بإدارة مشاريع تطوير الإنتاجية . ويمكن بمساعدتها جدولة أنشطة المشروع باستخدام الأساليب المعروفة مثل طريقة المسار الحرج Critical Path Method وأسلوب تقييم ومراجعة البرامج Programme Evaluation and Review Technique (PERT). كما تساعد في الوصول إلى التخصيص الأمثل للموارد المتاحة بما يضمن إنجاز نظام المعلومات في

اقصر وقت ممكن ضمن التكاليف المقررة . كما تعتبر هذه الأدوات مفيدة جداً لمتابعة سير العمل في المشروع وتقييم أداءه وإعداد التقارير الإدارية المختلفة .

4. مفهوم وأهمية هندسة المعلومات :

بالإضافة إلى التطورات المستمرة في أدوات هندسة البرمجيات بمساعدة الحاسوب ظهر في الثمانينات مفهوم جديد أطلق عليه اسم **هندسة المعلومات** Information Engineering، يطرح ضرورة البدء بتطوير أنظمة المعلومات انطلاقاً من الخطة الاستراتيجية للمنظمة . حيث يتم إعداد نموذج البيانات العام للمنظمة Corporate Data Model ، ثم تخطيط عمليات تطوير أنظمة المعلومات استناداً إلى هذا النموذج. أما أهم المزايا التي توفرها هذه المنهجية الجديدة فهي :

- **الاهتمام بالبيانات** من خلال تحديد النموذج العام لبيانات المنظمة ، الذي يدعم عملية اتخاذ القرارات فيها .
- **التوجه نحو تحقيق أهداف المنظمة** : حيث تبدأ هندسة المعلومات بتحديد الأهداف العامة للمنظمة، وبناء على هذه الأهداف يتم تحديد أهداف المستويات الإدارية الأخرى ، وتخطط مشاريع تطوير أنظمة المعلومات لتساعد في تحقيق الأهداف العامة للمنظمة وأهداف المستويات الإدارية المختلفة فيها.
- **استخدام أساليب النمذجة البيانية** لتمثيل البيانات وعرض استراتيجيات إدارة هذه البيانات واستراتيجيات إدارة المنظمة في نموذج واحد .
- **التوجه نحو المستخدم** ، حيث تستقطب هندسة المعلومات جميع الخبرات الموجودة في المنظمة للمشاركة الفعالة في تطوير نموذج البيانات العام

للمنظمة . بهدف تحديد البيانات والمعلومات اللازمة لاتخاذ القـــرارات في جميع المستويات الإدارية .

- **تعتمد هندسة المعلومات المنهجية التدرجية**، حيث يتم تطبيقـــها مـــن خلال سلسلة من الخطوات الرسمية التي يتم من خلالها التوســـع بنمـــوذج البيانات وإغناء تعاريفه بشكل مستمر خلال عملية التطوير .

- تساعد هندسة المعلومات في **تكثيف البيانـــات** Data Consolidation : وهذا يشمل **التدقيق المتبادل** Cross-Checking للكشف عـــن البيانـــات المتكررة . وفي النهاية يتم التوصل إلى نموذج متكامل للبيانات .

- توفر منهجية هندسة المعلومات **تغذية عكسية سريعة** ، لتمكين المنظمـــة من تحديث نموذج بياناتها في ضوء التغييرات الممكنة في الخطط الاستراتيجية نتيجة للظروف البيئية المختلفة.

- توفر منهجية هندسة المعلومات **إمكانية إجراء التحليـــلات المختلفـــة** للخطط الاستراتيجية وتحديد تأثير البدائل المختلفة على النمـــوذج العـــام لبيانات المنظمة، وتوفير المرونة الكافية لإدخال التعديلات اللازمـــة فيـــها بالسرعة الكافية .

- توفر منهجية هندسة المعلومات **بيئة تطوير محوسبة تعتمد على استخدام الأنظمة الخبيرة في التخطيط الاستراتيجي** وتطوير قاموس خبـــير يمكـــن تحويل تعريفاته بصورة تلقائية إلى خطط ومشاريع لتطوير أنظمة المعلومات اللازمة لدعم هذه التوجهات الاستراتيجية. ثم يتم ترجمة هـــذه الخطـــط بدورها أيضاً بشكل تلقائي وتوليد قواعد البيانات اللازمة لهذه الأنظمـــة .

وهكذا فإن هندسة المعلومات تعتمد على استخدام الأنظمة الخبيرة في عمليات تطوير أنظمة المعلومات .

1.4 مراحل هندسة المعلومات :

تهدف هندسة المعلومات إلى ربط عمليات تطوير أنظمة المعلومات بالتخطيط الاستراتيجي للمنظمة . ولذلك فإنها تتكون من المراحل الثلاث الرئيسية التالية :

1.1.4 مرحلة التحليل Analysis Phase :

في هذه المرحلة يتم تحديد إحتياجات المنظمة للبيانات في ضوء خططها الاستراتيجية والتكتيكية . وهنا يكون الدور الرئيسي للمستخدم باعتباره خبير في مجال الأعمال الذي تمارسه المنظمة . وحيث يقوم المستخدمين بالاستفادة من أساليب هندسة المعلومات الخاصة بنمذجة الخطط الاستراتيجية والتكتيكية ، ويستعينون بالنظم الخبيرة لأتمته عمليات التحليل (تحليل الخطط الاستراتيجية) . وتكون مخرجات هذه المرحلة عبارة عن نماذج بيانات المنظمة على المستويات الاستراتيجي والتكتيكي والعملياتي كما هو مبين في الشكل (2.15).

2.1.4 مرحلة التصميم Design Phase :

يتم خلال هذه المرحلة تحديد البيانات المتكررة، وترتيب وتنظيم البيانات وربطها (تكاملها) على مستوى المنظمة . وبحيث تستطيع جميع الأقسام في المنظمة الوصول إلى البيانات اللازمة لها ضمن صلاحيات محددة . وتتم هذه المرحلة بصورة آلية من خلال نظام خبير يقوم باستخلاص أو اشتقاق نماذج فرعية للبيانات Data Submodels من النموذج المتكامل لبيانات المنظمة الذي تم توليده في المرحلة

السابقة. وتسمى هذه النماذج الفرعية قواعد بيانات موضوعية Subject Databases. ويتم تحديد أنظمة المعلومات أو الأنظمة الخبيرة التي يمكن أن تساعد في تحقيق الأهداف الاستراتيجية بواسطة برمجيات خاصة تقوم بتقييم السياسات والاستراتيجيات والأداء الإداري في المنظمة وتحديد أولويات الأنظمة المختلفة. وفي النهاية يتم إعداد خطة لتنفيذ أنظمة المعلومات اللازمة وفق الأولويات المحددة ويتم تصميم مكونات هذه الأنظمة في ضوء الخطة المقررة .

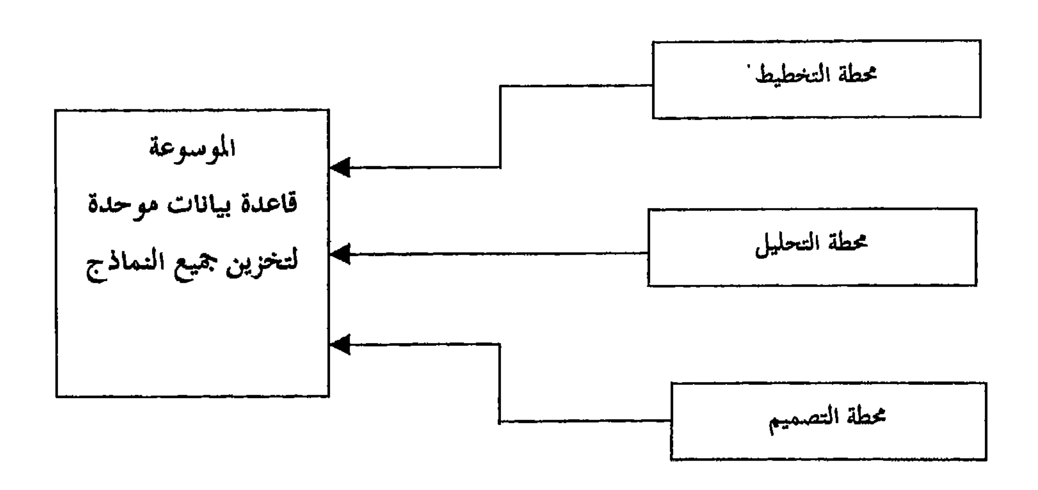
3.1.4 مرحلة التوليد Generation Phase :

ويتم خلالها توليد قواعد البيانات اللازمة لأنظمة المعلومات التي يجري تطويرها ، وكذلك بقية مكونات هذه الأنظمة من واجهات استخدام وبرمجيات معالجة وغيرها . وكل ذلك يمكن أن يتم بصوره آلية . كما يمكن في هذه المرحلة أيضا توليد أنظمة خبيرة في مجال إدارة النشاط الذي تقوم به المنظمة . وفي هذه الحالة يتم اشتقاق وصياغة قواعد النظام الخبير من الاستراتيجيات المحددة من قبل المستخدمين . وهكذا يتم بناء قاعدة معرفية في مجال إدارة الأعمال المعني بصورة آلية أيضا.

2.4 محطات عمل هندسة المعلومات Information Engineering Workbench:

تعتمد منهجية هندسة المعلومات على استخدام محطات عمل خاصة، يطلق عليها محطات عمل هندسة المعلومات (IEW) التي تم تطويرها من قبل جيمس مارتن مؤسس هذه المنهجية التي توفر مدخلا أكثر هيكلية يمكن من خلاله استخدام أدوات تطوير عديدة بطريقة مرنة . وتدعم محطات العمل هذه تجميع أدوات التطوير بحسب مراحل دورة حياة تطوير الإنتاجية . تتكون محطات عمل هندسة المعلومات من ثلاث محطات عمل كما هو مبين في الشكل (3.15).

شكل (2.15) المراحل الثلاث لهندسة المعلومات



شكل (3.15) مكونات محطة عمل هندسة المعلومات

توفر كل محطة من المحطات المبينة في الشكل أعلاه أدوات رسم المخططات المناسبة للمرحلة التي تمثلها . ويمكن استخدام محطة التخطيط لتطوير استراتيجية المنظمة ، أما محطة التحليل فتستخدم لدعم مهام وأنشطة تحليل الأنظمة ، بينما تستخدم المحطة الخاصة بالتصميم لدعم أنشطة التصميم التفصيلي لأنظمة المعلومات.

أما المحطات فتتألف بدورها من عدد من الأدوات وتقوم كل أداة بدعم أحد أنواع مخططات النظام . ويتم تخزين المعلومات المتعلقة بهذه المخططات في قاعدة متكاملة للبيانات تسمى الموسوعة Encyclopedia .

وبفضل استخدام هذه الموسوعة المركزية يتم القيام بالربط المتبادل Cross-Correlate للمعلومات المخزنة في المخططات المختلفة . وعمليا يمكن العمل باستخدام أحد هذه المخططات ثم فتح أنواع المخططات الأخرى انطلاقا من هذا

المخطط، فيتم عرض بيانات المخطط الأول وفق الطريقة المستخدمة في المخطط الذي تم فتحه (نوافذ متعددة).

تقوم فلسفة محطات عمل هندسة المعلومات على توفير دعم قوي في مجال رسم المخططات ، والربط المتبادل فيما بينها للمساعدة في صيانتها بشكل جيد . بينما تترك الحرية للمصمم لاختيار وتحديد مراحل تطوير النظام ، ومتطلبات التقارير الإدارية اللازمة لمتابعة العمل . وسنقدم فيما يلي وصفاً موجزاً لهذه المحطات الثلاث:

1) **محطة التخطيط The Planning Workbench :**

توفر هذه المحطة مجموعة من الأدوات التي تساعد في إعداد الخطط الأولية لأنظمة المعلومات وتحديد متطلبات الاستخدام الخاصة بها . أما أهم أدوات هذه المحطة فهي :

- **أداة رسم المخططات الجدولية Matrix Diagrammer** : وتقوم بتسجيل ومسك الترابطات المتبادلة بين البيانات والوظائف التي تستخدمها هذه البيانات .

- **أداة رسم المخططات Decomposition Diagrammer** : وتستخدم لنمذجة عمليات النظام وتحليلها إلى مكوناتها التفصيلية .

- **أداة رسم مخططات الكينونة Entity Diagrammer** : وتساعد في إعداد مخططات الكينونات المستخدمة في نمذجة البيانات .

ومن خلال مخرجات هذه المحطة يتم توصيف متطلبات تطوير أنظمة المعلومات ، وإعداد النموذج العام للمنظمة الذي يبين وظائفها وبياناتها .

2) **محطة التحليل The Analysis Workbench :**

تساعد هذه المحطة في حوسبة عمليات تحليل نظام العمل في المنظمة ، وذلك بالتركيز على مخططات تدفق البيانات (DFD) ، ونمذجة الكينونات، ولذلك فهي تحتوي على أداتين رئيسيتين هما **أداة رسم مخططات تدفق البيانات وأداة رسم مخططات الكينونات** . وبالإضافة إلى ذلك فإن محطة التحليل تحتوي على أداة **لرسم مخططات العمليات Action Diagrammer** وغيرها من مخططات التحليل .

وبشكل عام فإن جميع المخططات التي يتم إعدادها باستخدام هذه المحطة شديدة الصلة ببعضها البعض نظراً لكونها تمثل نفس النظام بالتركيز في كل مرة على أحد جوانبه أو أحد خصائصه الرئيسية . ولذلك يمكن للمحلل البدء بالعمل مع أحد هذه المخططات ثم الانتقال منها لفتح المخططات الأخرى ذات العلاقة . فمثلاً يمكن أن يختار المحلل عملية معينة في مخطط تدفق البيانات ثم " يفتح " منها نافذة تعرض التوصيف التفصيلي لهذه العملية والموجود في مخططات العمليات.

3) **محطة التصميم The Design Workbench :**

تدعم هذه المحطة عمليات التصميم وتستخدم لتحويل النماذج التي تم إعدادها بواسطة محطة التحليل إلى نظام معلومات جاهز للتشغيل . أما أهم أدوات هذه المحطة فهي أدوات رسم المخططات الهيكلية ، وأداة رسم العمليات وأدوات تخطيط العروض Presentation Diagrammer وتصميم الملفات ، وأدوات التحليل العلاقاتي وغيرها .

ويمكن استخدام هذه الأدوات لتوليد البرامج التنفيذية الخاصة بقاعدة البيانات وواجهات الاستخدام وغيرها من مكونات النظام .

3.4 تسهيلات هندسة المعلومات Information Engineering Facility (IEF):

تم تصميم هذه الخدمة لمساعدة محللي ومصممي أنظمة المعلومات ، وهـــي تحتوي على مجموعة أدوات Tool set وتستخدم خلال مرحلـــة التحليـــل لرســـم المخططات البيانية التالية : مخطط الكينونــــات - العلاقــات والهيكـــل الهرمـــي للكينونات، والمخطط الهرمي لتحليل العمليات ومخطط العمليات ، وأخيراً مخطـــط العمليات - الأعمال . ثم يتم تحويل هذه المخططات بشكل آلي إلى مجموعـة أدوات التصميم التي تحتوي ثلاث أدوات أخرى هي : أداة رسم مخططات الحـــوار ، وأداة تصميم الشاشات وأداة رسم مخططات إجراءات العمل . وتستخدم هــذه الأدوات لتصميم وإنتاج مكونات النظام من قاعدة بيانات وشاشات وبرامـــج تطبيقـــات وغيرها .

أسئلة الفصل

1- ما هي الفلسفة التي يستند إليها التوجه نحو حوسبة عمليات تطوير أنظمة المعلومات ؟

2- اشرح مزايا استخدام أدوات هندسة البرمجيات بمساعدة الحاسوب ؟

3- ما هي الأهداف الرئيسة الثلاث لهندسة البرمجيات ؟

4- كيف يمكن تصنيف أدوات هندسة البرمجيات بمساعدة الحاسوب ؟

5- ما هي أهم أدوات هندسة البرمجيات بمساعدة الحاسوب ، المستخدمة في مرحلتي التحليل والتصميم ؟

6- ما هي أهم أدوات هندسة البرمجيات بمساعدة الحاسوب المستخدمة في مرحلتي التنفيذ والصيانة ؟

7- ما المقصود بالأدوات المتكاملة لهندسة البرمجيات بمساعدة الحاسوب ؟ وما هي ميزاتها ؟

8- اشرح المكونات الرئيسة لأدوات هندسة البرمجيات بمساعدة الحاسوب ؟

9- اشرح الطرق التي يمكن من خلالها تحقيق الاستخدام المتكامل لأدوات هندسة البرمجيات بمساعدة الحاسوب ؟

10- ما هي هندسة المعلومات وما هي أهم مزايا هذه المنهجية الجديدة لتطوير الأنظمة ؟

11- اشرح المراحل الثلاث لهندسة المعلومات ؟

12- عدد واشرح محطات عمل هندسة المعلومات ؟

13- اشرح أهم الأدوات التي تتضمنها محطات التخطيط والتحليل والتصميم في هندسة المعلومات ؟

14- ما هي تسهيلات هندسة المعلومات ؟

الفصل السادس عشر

إدارة مشروعات
تطوير أنظمة المعلومات

يعالج هذا الفصل بشكل شمولي جميع القضايا الإدارية المتعلقة بتطوير أنظمة المعلومات. فبالإضافة إلى عمليات التحليل والتصميم التي تمت دراستها خلال فصول هذا الكتاب، تتضمن عملية التطوير مراحل أخرى هي مرحلة التنفيذ ومرحلة ما بعد التنفيذ. ويقتصر دور المحلل أو المصمم فيها على الإشراف والتنسيق والتوجيه والمتابعة لضمان تنفيذ النظام الجديد وفق التصاميم والأهداف المقررة، وضمن الفترة الزمنية المحددة وبالتكلفة المناسبة. أن عملية التطوير بما تتضمنه من أنشطة ومهام عديدة متنوعة، ولما تتطلبه من موارد مادية وتقنية وبشرية، تحتاج إلى تخطيط وتنظيم ومتابعة ورقابة لتنسيق هذه الأنشطة وتوجيهها نحو تحقيق أهدافها المنشودة. وهذا ما نطلق عليه **إدارة المشروعات**، فعمليات تطوير الأنظمة هي عبارة عن مشروعات حقيقية يجري خلالها استخدام أنواعاً مختلفة من الموارد لبناء أنظمة المعلومات اللازمة لتلبية احتياجات المنظمة. ولذلك فإن استخدام مفاهيم وأساليب إدارة المشاريع يعتبر أمراً أساسيا لنجاح عملية تطوير هذه الأنظمة، سندرس في هذا الفصل جميع القضايا الإدارية المتعلقة بعمليات التطوير والتي من خلالها يمكن توفير عوامل النجاح الحرجة واللازمة لتطوير أنظمة المعلومات.

1-إدارة الجودة الشاملة في مشاريع تطوير أنظمة المعلومات

يعتبر موضوع الجودة من أهم الموضوعات التي يجب أن يتعامل معها فريق تطوير أنظمة المعلومات. ويقصد بالجودة مطابقة النظام الجديد للمتطلبات أو الخصائص التي يرغبها المستخدم، فالنظام الجيد هو الذي يلبي متطلبات المستخدم بكفاءة، وليس ذلك النظام الذي يستخدم احدث التقنيات أو الأساليب لمجرد كونها أدوات حديثة وحسب.

1.1 خطوات إدارة الجودة:

يمكن تلخيص خطوات تطبيق مفاهيم إدارة الجودة الشاملة خلال مراحل تطوير نظام المعلومات على النحو التالي:

أ) **خلال مرحلة التحليل:** يجب خلال هذه المرحلة التعرف على مستخدمي النظام وتحديد فئاتهم المختلفة، ثم تحديد متطلباتهم فيما يتعلق بجودة المعلومات وجودة النظام الذي يستخدمونه، أي أنه في هذه المرحلة يجب تحديد الخصائص التي يرغب المستخدمون أن تتوفر في النظام الجديد الذي يجري تطويره، والتي يمكن أن تشمل:

- **الدقة Accuracy المطلوبة في عمليات النظام:** مدخلاته وعملياته ومخرجاته.
- **الموثوقية Reliability** في المخرجات من تقارير واستعلامات.
- **سهولة استخدام النظام User-Friendliness** والتعلم عليه ومعالجة مشاكله المختلفة.
- **الأداء Performance:** أي مستوى أداء الوظائف المختلفة التي يقوم بها النظام.

- **سرعة استجابة النظام** Responsiveness للأحداث المختلفة.
- **المرونة أو سهولة التعديل:** ويقصد بها توفر المرونة الكافية لإجراء بعض الإضافات أو التغييرات في النظام لضبط مخرجاته وفق الاحتياجات المتغيرة في البيئة.

وبعد تحديد خصائص الجودة المطلوب توفرها في النظام الجديد يجب أن يتم تحديد كيفية قياس هذه الخصائص، ويفضل هنا الاتفاق على معايير جودة Quality Standards قابلة للقياس الكمي، مثلاً أن يكون زمن الاستجابة لا يتعدى ثلاثة ثواني، أو يكون معدل الأداء ترحيل ستون فاتورة في الدقيقة وهكذا.

ب) خلال مرحلة التصميم:

في مرحلة التصميم العام يجري توليد عدد من البدائل المختلفة لتصميم النظام الجديد، ويسبق ذلك تحديد استراتيجيات الجودة اللازمة لتحقيق الأهداف. وهكذا يجب توليد الحلول البديلة لتصميم النظام الجديد في ضوء أهداف النظام ومتطلبات المستخدم وكذلك الاستراتيجيات المختلفة للوصول إلى الجودة المطلوبة. أما هذه الاستراتيجيات فيمكن أن تتضمن:

- الاهتمام بالتوظيف والتدريب وتنمية الموارد البشرية المتخصصة في أنظمة المعلومات.
- الاهتمام بالمستخدمين والعمل على تلبية احتياجاتهم بأفضل السبل الممكنة.
- استخدام المنهجيات العلمية لتحليل العمليات والمهام التي يقوم بها النظام، وتحديد معايير قبول النظام الجديد استناداً إلى مستويات أداء هذه العمليات.

- استخدام أسلوب النماذج التجريبية للتوصل إلى تحديد افضل لمتطلبات المستخدم.
- إجراء الاختبارات المختلفة لقياس مستوى أداء النظام.

وهكذا فأنه في مرحلة التصميم يتم تحديد الاستراتيجيات التي سيتم استخدامها لتحقيق خصائص الجودة التي تم تحديدها في مرحلة التحليل.

جـ) في مرحلة التنفيذ: تركز إدارة المشروع في هذه المرحلة على التأكد من تطبيق الاستراتيجيات التي تم تحديدها خلال مرحلة التصميم.

د) في مرحلة ما بعد التنفيذ: تقوم إدارة المشروع بتقييم جودة النظام من خلال قياس الجودة الفعلية للنظام ومقارنتها مع المعايير المحددة خلال مراحل التحليل والتصميم، ويمكن قياس الجودة الفعلية من خلال دراسة المعلومات التي ينتجها النظام والتعرف على آراء المستخدمين، ومدى استفادتهم منه وتأثيره على الأعمال التي يقومون بها.

2.1 أساليب ضمان الجودة Quality Assurance Techniques

تستخدم هذه الأساليب خلال مراحل دورة حياة تطوير نظام المعلومات للتأكد من أن النظام الجديد سوف يحقق متطلبات مستخدميه، وسوف يتم إنجازه وتسليمه في الموعد المحدد، وضمن الموارد المخصصة له. وتعتمد هذه الأساليب بشكل عام على تقييم أداء النظام بالمقارنة مع متطلبات استخدامه، والتأكد من عدم وجود أخطاء في تصميمه (في وثائق التصميم) أو تنفيذه، بالإضافة إلى تحديد المزايا الأخرى كسهولة الاستعمال والمرونة وإمكانية النقل Portability من بيئة تشغيل إلى أخرى، والتأكد من قابليته للاستخدام ضمن بيئة العمل الموجودة في المنظمة، أما أهم الأساليب التي يمكن اتباعها للتأكد من جودة نظام المعلومات فهي:

1.2.1 التفتيش أو التدقيق Inspections

وهو ابسط الأساليب المستخدمة للتأكد من الجـــودة، وتتـــم المراجعـــة أو التدقيـــق من قبل فريق يتكون من صاحب النظام (المدير، أو المستخدم الرئيســـي) والمفتش أو المدقق. ومن المهم جداً أن لا يضم فريق التفتيش أي مـــن الأشـــخاص الذين شاركوا في التحليل والتصميم، ويمكن أن تتم عمليات التفتيش بعد إنجاز كل مرحلة من مراحل دورة حياة النظام، للتأكد من جودة تنفيذها قبل الانتقـــال إلى المرحلة التالية، ويجب أن تكون عمليات التدقيق ذات طابع رسمي وتنتهي بـــإعداد تقرير يتضمن توصيات واقتراحات لتحسين جودة النظام، ويجب أن يتعامل فريـــق التطوير بجدية تامة مع هذا التقرير في كل مرحلة من مراحل دورة تطوير النظـــام، وأن يقوم بتنفيذ ما يتضمنه من توصيات.

ومن الضروري أيضاً أن يقوم فريق التدقيق بوظائف المتابعة للتأكد من تنفيذ التوصيات وإزالة نقاط الضعف والقصورات المكتشفة خلال عمليـــات التفتيـــش والتدقيق.

2.2.1 المراجعات Walkthroughs

تستخدم المراجعات بشكل واسع في عمليات إدارة الجودة للتأكد من صحة النماذج الموضوعة من قبل المحللين والمصممين، ويمتاز هذا الأسلوب عن التفتيـــش بكونه يعتبر جزءاً من المنهجيات الهيكلية Structured Methodologies المســتخدمة في تطوير الأنظمة، ولذلك يســـمى أيضـــاً بالمراجعـــات الهيكليـــة Structured Walkthrough. ويقترح هذا الأسلوب إجراءات محددة لتدقيق ومراجعـــة جميـــع عناصر ومكونات نظام المعلومات، كما يتطلب فريق مراجعة متخصصـــص، يقـــوم

كل عضو من اعضاءه بمهام محددة، ويتم خلالها إعداد وثائق المراجعة، فمثلاً يجب أن يدقق فريق المراجعة النقاط التالية في نموذج النظام:

- تحقيق أهداف النظام وتلبية متطلبات مستخدميه.
- صحة تمثيل النموذج للنظام الأصلي.
- عدم وجود أي نقص، أو حذف أي نقاط ضرورية وعدم وجود أي غموض أو ابهام فيه.
- إمكانية قيام النظام بالوظائف المطلوبة منه.
- سهولة فهم النظام.

ويجب تنظيم عمليات المراجعة خلال جميع مراحل دورة حياة النظام، حيث يمكن أن تبدأ بعد الإنتهاء من إعداد النموذج المادي والنموذج المنطقي للنظام الحالي. وتهدف هذه المراجعة إلى التأكد من أن هذه النماذج تعبر بدقة وبشكل صحيح ومكتمل عن النظام الحالي و لم تهمل أية جوانب أو عناصر هامة فيه.

وكذلك الأمر بعد الإنتهاء من إعداد النموذج المنطقي للنظام الجديد يجب القيام بمراجعة ثانية للتأكد من عدم وجود أية أخطاء أو نقاط ضعف أو عدم إغفال أي عنصر فيه. وكذلك يجب تكرار عملية المراجعة هذه بعد الانتهاء من إنجاز النموذج المادي للنظام الجديد، ومن الطبيعي أن هذه المراجعات يجب أن تنتهي بتقارير موثقة، وأن يتم متابعة تنفيذ المقترحات والتوصيات الموجودة فيها.

تتكون إجراءات المراجعة من أربعة خطوات رئيسه هي:

أ) التحضير للمراجعة:

بعد أن يتأكد المحلل أو المصمم من أن النموذج الذي يطوره قد اصبح مكتملاً وجاهزاً للمراجعة، يمكن دعوة فريق المراجعة للاجتماع وإسناد

أدوار محددة لكل منهم، ثم يتم توزيع الوثائق على أعضاء الفريق وإعطاءهم الوقت الكافي لدراستها والتعرف على ما تتضمنه من حلول، وبعد ذلك يمكن تحديد موعد ليجتمع الفريق للقيام بعملية المراجعة.

ب)تنفيذ عملية المراجعة.

خلال جلسة أو جلسات المراجعة، يقوم المحلل أو المصمم الذي وضع نموذج النظام بشرح تفاصيل عمل النموذج من خلال عرض المخططات والوثائق المختلفة، ويقوم أعضاء فريق المراجعة، كل حسب دوره، بإبداء الملاحظات والتعليقات المتعلقة بنقاط الضعف الموجودة في النموذج وتحديد المواضع التي تتطلب دقة أو إيضاح اكثر، أو تلك التي يجب فيها إدخال إضافات جديدة في النموذج، ويتم تسجيل ذلك في قائمة ليتم متابعة تنفيذها فيما بعد.

جـ) توثيق عمليات المراجعة:

من الضروري جداً توثيق نتائج المراجعة كوسيلة لمتابعة تنفيذ التصحيحات والتعديلات اللازمة في النموذج. ويتم عادة توثيق نتائج المراجعات في وثيقتين: الأولى تقرير ملخص يصف النتائج التي تم التوصل إليها خلال المراجعة. والوثيقة الثانية عبارة عن قائمة الأعمال أو الإجراءات التي يجب القيام بها في ضوء نتائج المراجعة، ويجب أن تحدد هذه القوائم المشكلات التي يجب معالجتها والمسؤولين عن إيجاد الحلول المناسبة لهذه المشكلات.

د) المتابعة:

في هذه الخطوة يتم متابعة تنفيذ توصيات فريق المراجعة بالاستناد إلى قائمة الإجراءات التي أشرنا إليها في الخطوة السابقة. ومن الواضح أن هدف المتابعة هو التأكد من حل المشكلات التي تم اكتشافها خلال عمليات المراجعة.

وتجدر الإشارة أخيرا إلى أن إجراء مراجعات ناجحة يتطلب أولا أن يكون أعضاء الفريق الذي يقوم بالمراجعة ذوي خبرة عالية نظرياً وعملياً، كما يجب أن يكونوا موضوعين ومحايدين قدر الإمكان وأن يقدموا أفكار ومقترحات بناءة وأن يبتعدوا عن المناقشات والمهاترات والدخول في السلوك الجدلي الذي لا فائدة منه. كما يجب أن يتجنبوا أسلوب النقد أو الهجوم أو التقييم في تعاملهم مع فريق التطوير، فالهدف من المرجعة هو إدخال التحسينات في النظام وليس إثبات ضعف أو فشل أو تقصير فريق التطوير.

وتجدر الإشارة أيضاً إلى ضرورة إبراز النقاط الجيدة في النموذج الذي تجري مراجعته مما يساعد في الحكم الموضوعي على تصميمه، ويجذب فريق التطوير للتعاون الفعال مع فريق المراجعة. ومن المهم أيضا الإشارة إلى أن على المراجعين أن يكتشفوا المشاكل ويثيروا التساؤلات حول النقاط المختلفة للنموذج، وليس مطلوب منهم بأي حال من الأحوال أن يضعوا الحلول لهذه المشاكل والتساؤلات، ومن ناحية أخرى على أعضاء فريق المراجعة محاولة الاستيضاح عن الجوانب الغير مفهومه لهم في النموذج بدلاً من إبداء الانتقادات بشأنها.

2. إدارة عمليات التنفيذ والتقييم والصيانة

بعد الإنتهاء من تحليل وتصميم النظام ومراجعة التصاميم التفصيلية له والتأكد من جودتها ومطابقتها لمتطلبات الاستخدام ولأهداف النظام، تبدأ المرحلة التالية في دورة حياة تطوير النظام، وهي مرحلة التنفيذ أي تحويل هذه التصاميم إلى نظام معلومات يقوم بوظائفه وفقاً للمتطلبات التي تم تحديدها في مرحلة التحليل. وبعد وضع النظام قيد التشغيل لفترة زمنية مناسبة يصبح من الضروري

تقييم الأداء الفعلي لهذا النظام، للتعرف على مدى مطابقته للمعايير المحددة في متطلبات الاستخدام، وتسمى هذه العملية **مراجعة وتقييم النظام**.

ويمكن أن تؤدي هذه العملية إلى إدخال بعض التعديلات في النظام. وبشكل عام يحتاج النظام خلال فترة تشغيله واستخدامه إلى عمليات إدامة أو **صيانة Maintenance** يتم خلالها إدخال التعديلات الضرورية فيه والناتجة عن تغير متطلبات المستخدم نتيجة التغيرات المختلفة التي يمكن أن تحدث في بيئة عمل المنظمة.

وهكذا فإن علاقة محلل ومصمم النظام مع نظام المعلومات لا تنتهي بانتهاء مرحلتي التحليل والتصميم، بل تستمر خلال جميع مراحل دورة حياته، ولكنها تتغير من حيث الدور الذي يقوم به في كل مرحلة. وفي مرحلتي التنفيذ وما بعد التنفيذ يكون دور المحلل أو المصمم كما ذكرنا إشرافياً أي إداريا، حيث يكون مسؤولا عن تخطيط وتنظيم ورقابة جميع الأنشطة التي تتضمنها وهذا ما سنتعرف عليه في هذه الفقرة.

1.2 إدارة أنشطة تنفيذ النظام

تتضمن مرحلة التنفيذ جميع الأنشطة اللازمة لتنفيذ نظام المعلومات الجديد ووضعه قيد التشغيل وهي:

- **البرمجة والاختبار:** وتتم خلالها كتابة برامج النظام واختبارها. تشمل هذه البرامج بناء قاعدة بيانات النظام وواجهات الاستخدام وتنفيذ جميع عمليات ووظائف النظام.
- **إعداد موقع النظام:** وهذا يشمل الحصول على التجهيزات اللازمة وتركيبها واختبار جاهزيتها للعمل.

- **تدريب المستخدمين** على استخدام النظام.
- **التحول** من النظام القديم إلى النظام الجديد.

1.1.2 إدارة عمليات البرمجة والاختبار:

يعتبر محلل أو مصمم النظام مسؤولاً عن تخطيط عمليات البرمجة والاختبار ومتابعة تنفيذها للتأكد من تطبيق مقاييس ومعايير الجودة الخاصة بها، والتي يضعها المصمم عادة في دليل خاص يسمى دليل المبرمج، ليكون مرشداً لجميع المبرمجين المشاركين في المشروع.

كما يقوم المصمم بوضع **خطة لعمليات البرمجة والاختبار**، يتم من خلالها تحديد الأسلوب أو المنهجية التي ستستخدم في هذه العمليات، ويمكن أن تكون هذه البرمجة تراكمية أي يتم كتابة البرنامج الخاص بالوحدة الوظيفية واختباره، ثم إضافة وحدة وظيفية أخرى واختبارها، ثم ربط الوحدتين معاً واختبار عملهما، ثم إضافة وحدة وظيفية ثالثة، وهكذا.

أما طريقة ربط هذه الوحدات واختبارها فيمكن أن تبدأ من الأعلى (الوحدة الوظيفية الرئيسية للنظام) ثم تمتد هبوطاً نحو الأسفل إلى الوحدات الوظيفية التالية وهكذا، وتسمى مثل هذه الطريقة Top-Down-Testing. كما يمكن أن تحدث بطريقة معاكسة أي تبدأ بالوحدات الوظيفية الموجودة في المستوى الأدنى ثم الانتقال منها نحو المستويات العليا. وأخيراً يمكن تنفيذ هذه العملية بواسطة فريقين يبدأ أحدهما من الوحدة الوظيفية الرئيسة ثم يتجه نحو المستويات الدنيا، بينما يبدأ الفريق الآخر من المستوى الأدنى ويتجه صعوداً نحو المستويات الأعلى.

وبالإضافة إلى خطة البرمجة والاختبار يقوم المصمم **بإنشاء البيانات اللازمة للاختبار**، والتي يجب أن تغطي جميع الاحتمالات والحالات التي يمكن أن يعالجها

النظام. ولذلك يجب تحديد بيانات لتجربة جميع القرارات والدورات وغير ذلك من البنى المختلفة الموجودة في هذه البرامج، كما يجب أن تتضمن بيانات الاختبار أيضاً استخدام قيم لحالات غير مقبولة أو غير معقولة لمعرفة رد فعل النظام وقدرته على اكتشافها.

أما أنواع الاختبارات التي يجب أن تتضمنها هذه الخطة فهي: اختبار الوحدات الوظيفية، ثم الاختبار التراكمي لهذه الوحدات، ثم الاختبار المتكامل للنظام، وأخيراً اختبارات قبول النظام التي يجري فيها التأكد من أن الأداء الفعلي للنظام ضمن المعايير المقررة، ويجب أن يتابع المصمم تنفيذ هذه الخطة ويعمل على توفير الموارد اللازمة لتنفيذها.

2.1.2 تدريب المستخدمين: User Training

على فريق التطوير أن يهتم بتدريب المستخدمين لإكسابهم المعارف والمهارات اللازمة لاستخدام النظام الجديد، إن إشراك المستخدمين بشكل فعال في مهام التحليل والتصميم، والرجوع إليهم، والتعرف على آرائهم، ومناقشة مشكلات النظام معهم، للوصول إلى حلول مناسبة لقرارات التصميم المختلفة، يعتبر جزءاً من عملية التدريب وإعداد المستخدمين للعمل في بيئة النظام الجديد. ولكن ذلك غالباً لا يكون كافياً فمن خلال المشاركة الفعالة يتكون لديهم فهم جيد للنظام ووظائفه وإمكاناته. وخلال مرحلة التنفيذ لا بد من تدريبهم على استخدام النظام وتعريفهم بكيفية الاستفادة منه في إنجاز الأعمال الموكولة إليهم، ولهذا الغرض يجب أن يقوم المصمم أو المطور بوضع خطة لتدريب المستخدمين يحدد فيها:

- **فئات المستخدمين المطلوب تدريبهم**، سواء اللذين سيتعاملون مع النظام بشكل مباشر أو غير مباشر، وهذا يشمل جامعي البيانات ومدخليـــــها والأفراد الذين سيوفر لهم أنواع الدعم المختلفة وغيرهم.
- **مصادر الحصول على التدريب**: أي تحديد مـــن ســيقوم بـــالتدريب المطلوب من داخل أو خارج المنظمة.
- **طرق التدريب**: وهنا يتم تحديد نوع التدريب المطلوب فمثلاً تدريــــب أثناء العمــــل On-Job-Training، أو تدريـــب خـــارجي في مراكـــز ومؤسسات التدريب المتخصصة، أو تدريب ضمن المؤسسة In -House Training أو غيرها.
- **أهداف التدريب**: لكل فئة من فئات المستخدمين، وهنا يجـــب تحديـــد المهارات العملية والمعارف العلمية التي يجب أن يتقنها ويحصــــل عليـــها المتدربون في نهاية البرنامج التدريــبي.

وعلى المحلل أو المصمم المسؤول عن تطوير النظام أن يتابع تنفيذ هذه الخطـــة في أوقاتها المحددة، لكي يتم تدريب المستخدمين قبل وضع النظام قيـــد التشـــغيل. وأخيراً تجدر الإشارة إلى أهمية **وثائق وأدلة استخدام وتشغيل النظام** ودورهـــا في تمكين المستخدم من التدرب بسهولة وسرعة على استخدام النظام، وباعتبارها تمثل مصدراً رئيسياً للمعلومات المتعلقة بالنظام يمكن للمستخدم الرجوع إليـــها كلمـــا دعت الحاجة.

3.1.2 تحويل النظام System Conversion

يقصد بهذه العملية التحول من العمل بالنظام الحالي إلى النظام الجديد الـــذي تم برمجته واختباره. ويتم خلال هذه العملية نقل البيانات من النظـــام الحـــالي إلى

النظام الجديد، وهذا يتطلب إنشاء الملفات (قاعدة البيانات) الـتي ستستخدم في النظام الجديد ونقل البيانات إليها، وتسمى هذه العملية تحويـل الملفـات وهـي تتطلب وقتاً وجهداً لذلك يجب التخطيط لها بعناية، وهذا يتضمن:

- تحديد مصادر البيانات التي سيتم إدخالها إلى النظام الجديـد، وكذلـك طريقة إدخال هذه البيانات (من خلال لوحة المفـاتيح، أو باستخدام الطرق الآلية).
- تحديد المواعيد الزمنية التي سيتم فيها تحويل البيانات من النظام القـديم إلى الجديد، والأفراد المسؤولين عن تنفيذ هذه العملية، وتوفـير المـوارد اللازمة لهم.

أن ضمان التحويل الناجح للنظام وبدون حدوث أي مشاكل أو تأخـيرات خلال هذه العملية، يتطلب من المحلل أو المصمم وضع خطة محكمـة، وأن يقـوم بمتابعتها بشكل كامل. والإدارة الجيدة لعملية التحويل يجب أن تعتمد على تخطيـط هذه العملية باستخدام الاستراتيجيات المناسبة لهذه العملية والتي يمكن أن تكون:

أ) التحويل المتوازي Parallel Conversion

وفقاً لهذه الاستراتيجية يتم البدء بتشغيل النظام الجديد مـع إبقـاء النظـام القـديم قيد التشغيل لفترة من الوقت، وريثما يتم التأكد من صحة تنفيذ النظـام الجديد وعدم وجود مشاكل فيه. وتستخدم هذه الاستراتيجية بشـكل واسـع في عملية التحويل نظراً لما توفره من إمكانية مقارنة نتائج عمـل النظـامين القـديم والجديد، وتقييم النظام الجديد ولما توفره من شعور بالأمان وعدم المخاطرة في بداية عمليات التشغيل أما مساوئها فتتلخص في التكلفة العالية لها.

ب) التحويل التدريجي Gradual or Phased Conversion

وفقاً لاستراتيجية التحويل التدريجي يتم الانتقال من النظام القديم إلى النظام الجديد على مراحل Phases. ففي كل مرحلة يتم استبدال جزء من النظام القديم بالنظام الجديد. وتتيح هذه الطريقة مرونة كبيرة لمعالجة المشكلات التي يمكن أن تظهر خلال مرحلة التحويل، كما تساعد في تسريع الحصول على بعض فوائد النظام الجديد قبل الانتهاء من عمليات التحويل، أي في فترة مبكرة من مرحلة التنفيذ، أما مساوئ هذه الاستراتيجية فهي كونها تؤدي إلى إطالة فترة التنفيذ.

جـ) التحويل المباشر Direct Conversion

وفقاً لهذه الاستراتيجية يتم الانتقال من النظام القديم إلى النظام الجديد مرة واحدة وفي لحظة معينة (تاريخ محدد). وبالرغم من وجود مخاطر كبيرة في استخدام هذه الاستراتيجية إلا أنها يمكن أن تكون ناجحة إذا تم التحضير لها بصورة جدية. وهذا يشمل القيام باختبارات مكثفة لبرامج النظام قبل البدء بالتحويل، كما يمكن أن تستخدم هذه الاستراتيجية إذا كان نظام المعلومات الذي يجري تطويره يمكن أن يتحمل بعض التأخير في إنجاز العمليات في حال حدوث مشاكل في بداية عمليات التشغيل. ومن الواضح أن استخدام هذه الاستراتيجية يتطلب القيام بالاستعدادات اللازمة لها وتوفر مستوى عال من تنظيم الأعمال لكي تتم دون مشاكل. أما ميزاتها فهي كونها تساعد في تحفيز جميع العاملين في تطوير النظام ومستخدميه للمشاركة بفعالية وحماس لضمان نجاح النظام الجديد حيث لا مجال للتردد والتفكير بالعودة إلى النظام القديم.

وأخيراً تجدر الإشارة إلى أن اختيار الاستراتيجية المناسبة لتحويل النظام يجب أن تتم وفقاً لظروف وطبيعة وحجم كل مشروع وفي كل حالة على حدة، ولكنه لا ينصح بشكل عام إطالة فترة التحويل لما لذلك من تأثيرات سلبية على المشروع.

2.2 إدارة أنشطة التقييم والصيانة

بعد الانتهاء من تنفيذ النظام ووضعه قيد التشغيل لفترة زمنية معينة (من ستة أشهر إلى سنة) يجب تقييم الأداء الفعلي لهذا النظام لتحديد مدى مطابقته للأهداف التي تم تحديدها في بداية عملية التطوير، وكذلك التعرف على مدى قبوله من قبل المستخدمين ونجاحه في تلبية متطلباتهم، وبالتالي تقرير فيما إذا كان بحاجة إلى إدخال بعض التعديلات فيه أم لا.

كما يعتبر هذا التقييم مفيداً جداً لتجميع المعلومات اللازمة لصيانة النظام، فأي نظام للمعلومات ومهما كان مكتملاً وناجحاً عند تنفيذه، سيحتاج بعد فترة من الزمن إلى إدخال بعض التعديلات والتغييرات، وذلك لتلبية الاحتياجات الجديدة للمستخدمين التي تظهر غالباً نتيجة التغيرات المستمرة في بيئة النظام. ولذلك فان وظيفة إدامة أو صيانة النظام System Maintenance تعتبر ضرورية جداً لضمان استمرارية قدرة النظام على تلبية الاحتياجات المتجددة للمستخدمين.

1.2.2 مرحلة التقييم والمراجعة.

تهدف هذه المرحلة كما أشرنا أعلاه إلى قياس الأداء الفعلي للنظام ومدى تلبيته للأهداف المنشودة منه، وهذا يجب أن ينعكس في تحسين أداء المستخدمين، وقيام النظام بإنتاج المخرجات المطلوبة، وعند إجراء هذا التقييم يجب التركيز على النقاط التالية:

- **جودة مخرجات النظام:** وهذا يتعلق بدقة هذه المخرجات،وسهولة الحصول عليها، واكتمالها ومناسبتها للأغراض الموجهة لها.
- **سهولة استخدام النظام**، وطريقة معالجة الأخطاء التي يمكن أن تحدث أثناء إدخال البيانات.
- **ثقة المستخدمين بالنظام** ومدى إقبالهم على استخدامه والاستفادة منه، وتأثيره الإيجابي على أعمالهم.

ويجب أن ينظم مطور النظام هذه المراجعات ويشرف بنفسه عليها، وهي تتم عادة باستخدام الطرق المعروفة لتجميع المعلومات مثل المقابلات الشخصية والاستبيانات والملاحظة وغيرها. كما يمكن الاستفادة من الأساليب الأخرى لتجميع المعلومات مثل مسك سجل لأحداث النظام المختلفة Log of Events، يتم فيه تسجيل المشاكل التي تحدث أثناء استخدام النظام وكيفية معالجتها، ويعتبر مثل هذا السجل ذو قيمة كبيرة لما يتضمنه من معلومات عن الأداء الفعلي للنظام.

وغالباً تتم عملية المراجعة هذه من قبل لجنة تقييم خاصة تضم أعضاء محايدين من داخل وخارج المنظمة، كما تشمل هذه المراجعة تدقيق دراسة الجدوى الفعلية للنظام ومقارنتها مع الدراسات التقديرية التي تم إعدادها خلال مراحل التطوير السابقة، وتحديد أسباب الانحرافات إن وجدت.

2.2.2 صيانة النظام System Maintenance

تهدف عمليات الصيانة إلى إدامة النظام أي المحافظة على استمرارية قدرته على تلبية احتياجات المستخدمين بكفاءة عالية، ولذلك تتضمن عملية الصيانة إدخال التحسينات والإضافات اللازمة فيه لتلبية هذه الاحتياجات. ويجب أن يرافق عمليات الصيانة هذه تحديث وتعديل وثائق النظام لتعكس بصورة دائمة الحالة

الراهنة للنظام. وتعتبر عمليات صيانة النظام من الأنشطة الهامة في دورة حياة تطوير المشروع والتي تتطلب من محلل النظام قدراً كبيراً من التخطيط والتنظيم وتخصيص الموارد البشرية والمادية اللازمة لذلك. بالإضافة إلى إعداد النماذج التي يتم من خلالها تحديد تأثير التغييرات المتخلفة في النظام، والتي تمثل نقطة الانطلاق لتخطيط وتنفيذ عمليات الصيانة.

3. التخطيط الاستراتيجي لأنظمة المعلومات: المفهوم والأهمية

تحتاج منظمات الأعمال إلى تطوير العديد من أنظمة المعلومات اللازمة لتسيير وإدارة أنشطتها المختلفة، وغالباً تستخدم هذه المنظمات أنظمة المعلومات كأداة دعم لتوفير عوامل النجاح اللازمة لها لتحقيق أهدافها في بيئة تتصف بالمنافسة الشديدة والتغير الدائم. ونظراً للدور الهام لأنظمة المعلومات وللإمكانات الكبيرة التي توفرها في مجالات إدارة الأعمال المختلفة، يزداد التوجه في المنظمات إلى الاستفادة من هذه الأنظمة في تحسين عملياتها أو ما يطلق عليه حالياً **إعادة هندسة العمليات** **Process Reengineering**. وهذا يعني توظيف أنظمة المعلومات لإعادة تصميم أنظمة العمل المختلفة بطريقة أكثر كفاءة وفاعلية. ولقد أدى ذلك إلى تزايد الطلب في المنظمات على تطوير أنظمة المعلومات لتلبية الاحتياجات الوظيفية المختلفة وفي جميع المستويات. ولكن محدودية الموارد المعلوماتية في المنظمة لا تسمح بتطوير جميع الأنظمة المطلوبة دفعة واحدة، ولذلك يصبح من الضروري جداً الاهتمام بالتخطيط كوظيفة ضرورية لاختيار أنظمة المعلومات الأكثر أهمية وترتيبها وفق أولويات يتم تحديدها في ضوء الأهداف الاستراتيجية للمنظمة.

ونظراً لأن عملية التخطيط تغطي فترة مستقبلية يمكن أن تصل إلى عدة سنوات مقبلة، فإنها تتم على عدة مستويات هي:

- التخطيط الاستراتيجي ويمتد لفترة طويلة نسبياً تتراوح بين ثلاث وخمسة سنوات.

- التخطيط السنوي أو التخطيط قصير الأجل ويمتد لسنة واحدة.

وهكذا تقوم المنظمات بتخطيط أنشطتها من خلال مجموعة من الخطط الاستراتيجية والسنوية التي تتضمن الوسائل الكفيلة بتحقيق أهداف المنظمة.

ويتم تحديد أولويات أنظمة المعلومات في ضوء ارتباطها بالأهداف الاستراتيجية للمنظمة. ويتم تقسيم الخطة الاستراتيجية للمنظمة إلى خطط استراتيجية وظيفية تتعلق كل منها بوظيفة من وظائف المنظمة، كالإنتاج والتسويق والتمويل وأنظمة المعلومات، والموارد البشرية. وتكون الخطة الاستراتيجية لأنظمة المعلومات إحدى مكونات الخطط الاستراتيجية للمنظمة، وتهدف إلى ربط عملية تطوير أنظمة المعلومات في المنظمة بتحقيق الأهداف الاستراتيجية لها.

وهكذا يهدف التخطيط الاستراتيجي لأنظمة المعلومات إلى وضع الخطة الاستراتيجية لتطوير هذه الأنظمة، خلال فترة قادمة تتراوح بين ثلاث وخمسة سنوات، في ضوء الخطة الاستراتيجية العامة للمنظمة، ويجب أن تربط هذه الخطة أهداف أنظمة المعلومات التي سيتم تطويرها، بالأهداف الاستراتيجية للمنظمة. وبذلك يمكن ضمان تطوير أنظمة معلومات تساعد وتدعم تحقيق الأهداف الاستراتيجية.

تهدف الخطة الاستراتيجية لأنظمة المعلومات إلى صياغة إطار عام للتوجهات المستقبلية لتطوير هذه الأنظمة في المنظمة، ويجب أن تشمل هذه التوجهات تحديد المجالات المطلوب تطوير تطبيقات حاسوبية جديدة فيها، والمهام المطلوب القيام بها في هذا المجال، ثم تحديد الموارد اللازمة لذلك. كما يجب أن تتضمن هذه الخطة

استراتيجيات التغيير التي تنشدها المنظمة، وبشكل خاص فيما يتعلق بأنظمة المعلومات الحالية في ضوء المستجدات التقنية في هذا المجال، وفي ضوء التغيرات السريعة التي يمكن أن تحدث في بيئة العمل وتؤدي إلى ظهور متطلبات واحتياجات جديدة.

أن التخطيط الناجح لأنظمة المعلومات يجب أن يراعي النقاط التالية:

- ربط أهداف أنظمة المعلومات بالأهداف الاستراتيجية للمنظمة ورسالتها.
- التوجه للمستقبل أي استقراء البيئة الخارجية للمنظمة في المستقبل.
- الانطلاق من الاحتياجات الكلية للمنظمة وليس احتياجات جزء معين فيها.
- البحث عن الفرص والإمكانات التي يمكن أن تستفيد منها المنظمة لتحقيق أهدافها الاستراتيجية.
- اعتبار التخطيط عملية تفاعلية مستمرة Iterative.
- الاعتماد على الإبداع والعصف الذهني وإعداد السيناريوهات المختلفة لتوليد الحلول البديلة وتقييم تأثيرها على المنظمة.

وتجدر الإشارة هنا إلى أن تأثير أنظمة المعلومات على المنظمة يمكن أن يتم في اتجاهين، الأول هو توفير الدعم اللازم للوظائف والعمليات التي تقوم بها المنظمة لتتم بأعلى كفاءة ممكنة، والثاني هو في كون أنظمة المعلومات فرصة هامة للمنظمة تمكنها من تطوير منتجات وخدمات جديدة. ومعظم تطبيقات أنظمة المعلومات الحالية مصممة وفقاً للاتجاه الأول، حيث يتم توفير المعلومات اللازمة بالحجم والوقت والشكل المناسبين لتلبية احتياجات الوظائف والعمليات المختلفة.

وهذا يطرح مسألة هامة عند تخطيط أنظمة المعلومات حيث يجب العمل على الاستفادة منها في توفير خدمات ومنتجات جديدة تمكن المنظمة من تقوية موقعها التنافسي وتوثيق العلاقة مع زبائنها ومورديها وغيرهم من أطراف البيئة الخارجية لها.

2.3 خطوات التخطيط الاستراتيجي لأنظمة المعلومات:

يمكن أن تتم عملية التخطيط الاستراتيجي لأنظمة المعلومات في المنظمة وفق الخطوات التالية:

1) **دراسة بيئة المنظمة الخارجية** للتعرف على رسالتها وأهدافها الاستراتيجية والاستراتيجيات التي تعتمدها لتحقيق هذه الأهداف، وهنا يجب أن يتم تحديد عوامل النجاح الحرجة والتي يجب أن تكون هدفاً لتطوير أنظمة المعلومات فيها. وهذا يتضمن أيضاً مسح التطبيقات المختلفة لأنظمة المعلومات في هذه البيئة، سواء الموجودة حالياً أو تلك التي يجري تطويرها لتلبية الإحتياجات المستقبلية.

2) **دراسة البيئة الداخلية للمنظمة ولأنظمة المعلومات الموجودة فيها** بهدف تحديد نقاط القوة والضعف ومطابقتها مع الفرص والتهديدات التي يمكن توقعها في البيئة الخارجية، وهذا يسمح باستخدام نقاط القوة للاستفادة من الفرص وتجنب التهديدات.

وخلال هذه المرحلة يتم التعرف على واقع أنظمة المعلومات الحالية ومدى ارتباطها مع أهداف المنظمة ونجاحها في دعم تحقيق هذه الأهداف ويتم ذلك من خلال:

- تقييم استخدامات الأنظمة والموارد الحالية وكذلك التي يجري تطويرها حالياً، ومدى مساهمتها في تحقيق أهداف المنظمة، ويجب التركيز عند هذا التحليل على تحديد فعالية وكفاءة استخدام هذه الموارد والتوقعات المستقبلية بشأنها، وهذا يشمل دراسة حقيبة التطبيقات المحوسبة وتحديد نقاط القوة والضعف فيها.
- فحص البنية التحتية الحالية لأنظمة المعلومات وتحديد مدى ملاءمتها للاحتياجات المستقبلية، وهذا يشمل التقنيات الحاسوبية وتقنيات الاتصالات والبرمجيات بأنواعها والموارد البشرية وقواعد البيانات والطرق والإجراءات المستخدمة لتطوير أنظمة المعلومات.

3)وضع خطط أنظمة المعلومات:

يتم في هذه الخطوة إعداد الخطة الاستراتيجية لأنظمة المعلومات والتي تكون بشكل حقيبة لتطبيقات أنظمة المعلومات الضرورية لتحقيق الأهداف الاستراتيجية للمنظمة. ثم يتم في ضوء هذه الخطة الاستراتيجية وضع الخطط السنوية لتطوير أنظمة المعلومات في المنظمة.

3.3 مكونات الخطة الاستراتيجية لأنظمة المعلومات:

تعتبر الخطة الاستراتيجية لتطوير أنظمة المعلومات في المنظمة في غاية الأهمية لكونها تساعد في توفير إمكانية الاستفادة المثلى من الموارد المحدودة لتلبية الاحتياجات اللامحدودة إلى تطوير أنظمة المعلومات اللازمة لتحقيق الأهداف الاستراتيجية للمنظمة.

وفي ضوء هذه الخطة الاستراتيجية يتم التخطيط لمشاريع أنظمة المعلومات، حيث تصاغ الخطط السنوية كما ذكرنا من هذه الخطة الاستراتيجية في ضوء الموارد التي يمكن الحصول عليها أو توفيرها لهذا الغرض في كل عام.

أما أهم مكونات الخطة الاستراتيجية لأنظمة المعلومات فهي:

1)خطة تطوير أنظمة المعلومات:

تتضمن هذه الخطة أنظمة المعلومات الجديدة المطلوب تطويرها خلال سنوات الخطة، حيث تحدد **قائمة بهذه التطبيقات ومواعيد البدء والانتهاء من تطويرها وحجم الموارد اللازمة** لذلك، وكذلك علاقة هذه الأنظمة مع بعضها البعض ومع خطط عمل المنظمة ووحداتها المختلفة.

2)خطة إعادة هندسة الأنظمة الحالية:

تتضمن هذه الخطة المشاريع المتعلقة بإعادة هندسة أنظمة المعلومات الحالية التي تتطلب إجراء تغييرات كبيرة فيها وليس مجرد إضافة وحدات وظيفية جديدة أو تغيير في بنية سجلات معينة. وتظهر الحاجة إلى إعادة هندسة هذه الأنظمة نتيجة للتغيرات السريعة والمستمرة في قطاعات الأعمال المختلفة وفي التقنية، ويجب أن تتضمن هذه الخطة تحديد الأنظمة التي تحتاج إلى إعادة هندستها. والأهداف المنشودة من ذلك، بالإضافة إلى المواعيد الزمنية المخططة للبدء والانتهاء، وأخيراً الموارد اللازمة لذلك. وتستخدم في عمليات تحديث الأنظمة الحالية (إعادة هندستها) تقنيات هندسة البرمجيات مثل الهندسة العكسية Reverse Engineering التي يقصد بها تطوير نموذج النظام الحالي بناء على البرامج المصدرية المستخدمة فيه. ثم يجري إدخال التعديلات اللازمة لإعادة هندسة هذا النموذج وفقاً للمتطلبات الجديدة.

3) خطة تطوير قاعدة بيانات المنظمة.

تعتبر قاعدة البيانات مورداً هاماً من موارد المنظمة، فهي تحتوي عادة على حجم هائل من السجلات المتعلقة بالزبائن والمبيعات والمشتريات والموردين والموظفين والتسهيلات الإنتاجية المختلفة من آلات وتصاميم منتجات ومواصفات فنية وسجلات المخزون والسجلات المحاسبية المختلفة وغيرها.

ولذلك فإن قاعدة بيانات المنظمة تعتبر المكون الأكثر أهمية بالنسبة للمنظمة، فالتجهيزات عند تعطلها يمكن استبدالها، وكذلك البرمجيات يمكن شراءها ثانية، أما قاعدة البيانات فليس هناك من سبيل للحصول عليها لكونها تمثل عمليات المنظمة لسنوات طويلة، ولذلك يجب أن تتضمن الخطة الاستراتيجية للمنظمة:

أ) **خطة تطوير قاعدة البيانات** وفقاً لنموذج بيانات المنظمة الذي يمكن أن يوجد أيضاً في عدة مستويات: نموذج البيانات الخاص بخطة السنة الحالية، ونموذج البيانات الفعلي، ويجب أن تتوافق هذه الخطة مع خطط تطوير وتحديث أنظمة المعلومات المشار إليها في الفقرتين الأولى والثانية أعلاه.

ب) **خطة حماية قاعدة البيانات وضمان سلامتها وتوزيعها** على الحواسب المختلفة في المنظمة وتنظيم أسس وصلاحيات وصول الجهات المختلفة إليها واستخدامها.

4) خطة تأمين الموارد Strategic Resources Plan:

تتضمن هذه الخطة تحديد الموارد اللازمة لتنفيذ الخطط المشار إليها أعلاه. ويشمل ذلك التجهيزات الحاسوبية وشبكات الاتصالات والبرمجيات بأنواعها وتنمية الموارد البشرية اللازمة من أخصائيي معلومات وغيرهم.

4.3 تنفيذ الخطة الاستراتيجية لأنظمة المعلومات:

يتم تنفيذ الخطط المشار إليها أعلاه من خلال آلية الخطط السنوية، حيث توضع الخطط السنوية لتطوير الأنظمة الجديدة وتحديث الأنظمة الحالية وتطوير قاعدة البيانات وتوفير الموارد اللازمة لذلك. وتتضمن الخطط السنوية تفاصيل المشاريع المختلفة وتوزيع الموارد اللازمة لتنفيذها، ويتم متابعة تنفيذ هذه الخطط من خلال تتبع تنفيذ المشاريع المختلفة لكي تبدأ وتنهي بالأوقات المحددة لها وضمن التكاليف المقررة. وكذلك لكي يتم تنفيذها وفق التصاميم المعتمدة لضمان نجاحها في تحقيق أهدافها. ويمكن الاستعانة في إدارة مشاريع التطوير بأساليب إدارة المشروعات المعروفة مثل المخططات القضبانية Bar Charts (مخططات غانت) وطريقة المسار الحرج Critical Path Method وأسلوب تقييم ومراجعة المشروعات PERT وغيرها. وتساعد هذه الأساليب في تخطيط وجدولة ومتابعة المهام والأنشطة المختلفة المكونة لمشاريع تطوير أنظمة المعلومات، علماً بأنه تتوفر البرمجيات الجاهزة لدعم هذه العمليات الإدارية سواء بشكل حزم إدارة مشروعات Project Management Packages ، أو ضمن أدوات هندسة البرمجيات بمساعدة الحاسوب أو ضمن أدوات محطات عمل هندسة المعلومات التي تعرفت عليها في الفصل السابق.

أسئلة الفصل

1. ما المقصود بالجودة الشاملة وكيف يمكن تطبيقها خلال مراحل تطوير نظام المعلومات؟
2. ما هي أهم الخصائص التي يمكن من خلالها تحديد جودة نظام المعلومات؟
3. اشرح أساليب ضمان الجودة؟
4. اشرح مراحل عملية المراجعة الهيكلية؟
5. اشرح الأنشطة الرئيسية لمرحلتي التنفيذ وما بعد التنفيذ؟
6. ما هو دور محلل النظم في تدريب المستخدمين وفي عملية التحويل إلى النظام الجديد؟
7. قارن بين الاستراتيجيات المختلفة لتحويل النظام؟
8. ما أهمية التخطيط الاستراتيجي لأنظمة المعلومات؟
9. اشرح خطوات التخطيط الاستراتيجي لأنظمة المعلومات؟
10. اشرح مكونات الخطة الاستراتيجية لأنظمة المعلومات في المنظمة؟

المحتويات

صفحة

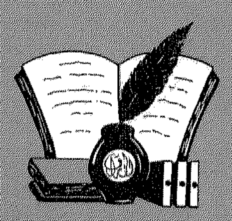
المراجع

1. Bellin D. And Suchman S., The Structured Systems Development Manual, Yourdon press, Prentice-Hall Building, Englewood Cliffs, 1990, p.221.
2. Dawns E., Clare P. And Coe I., Structured systems analysis and design method: application and context, second edition, prentice-Hall, New York, 1992.
3. Dixon R., Winning with CASE – Managing modern software development, McGraw-Hill Inc., New York, 1992, p.392.
4. Elmasri R. And Navathe S., Fundamentals of database systems, Second edition, the Benjamin / Cumming Pub. Co., Redwood city, 1994, p.872.
5. Finkelstein C., An introduction to information engineering from strategic planning to information systems, Addison-Wesley Publishing Co., Sydney, 1989, p.393.
6. Hawryszkiewycz I., System Analysis and Design, third edition, Prentice-Hall, Sydney, 1994, p.490.
7. Hipperson R., Practical systems analysis: a guide for users, managers and analysts, Prentice-Hall, New York, 1992, p.186.
8. Ince Darrel, Object – Oriented software engineering with c++, McGraw-Hill Int., Series in software engineering, London, 1991, p.231.

9. Kendall P., Introduction to systems analysis and design- a structure approach, second edition, WCB Publishers, 1992, p. 657.
10. Kendall K. and Kendall J. System analysis and design, second edition, Prentice-Hall, Englewood Cliffs, 1992, p.799.
11. Pressman R., Software engineering a practitioner's approach, third edition, McGraw-Hill Int. Edition, 1990, p793.
12. Senn J., Information systems for management, fourth edition, Wadsworth Publishing Co., Belmont, 1990, p.869.
13. Senn J. Analysis and design of information systems, second edition, McGraw-Hill Int. Edition, New York, 1989, p. 853.



System Analysis and Design



مؤسسة الوراق للخدمات الحديثة

نشر - توزيع - دعاية وإعلان

شارع الجامعة الأردنية - عمارة العساف - مقابل كلية الزراعة

تلفاكس (٥٣٣٧٧٩٨) - ص.ب (١٥٢٧) تلاع العلي

عمان (١١٩٥٣) الأردن

ISBN 9957 - 000 - 00 - 4 ردمك